TA 1600/30

TAXO

Bedienerhandbuch II

V 2.0.0

TN.337.99489-00

INHALTSVERZEICHNIS

0 EINLEITUNG

0.1 Grundlegende Charakteristika des System Command Interpreters
0.1.1 Definition der Terminals, an denen mit dem SCI gearbeitet werden kann
0.1.2 Login-Prozedur
0.1.3 Logoff-Prozedur
0.1.4 Privilegierung von Kommandos
0.1.5 Synonyme
0.1.6 Verzeichnis der TAXO SCI Kommandos

0.2 Kommandoformat und Darstellungsform
0.2.1 Aufrufzeichenkette
0.2.2 Kommandobezeichnung
0.2.3 Eingabeaufforderung
0.2.4 Typ der Eingabe
0.2.5 Eingabeformat

1 IDENTIFIKATION UND SCHUTZ VON DATEIEN

1.1 Allgemeines
1.2 Der Gebrauch von Pfadnamen (Pathnames)
1.2.1 Zusammensetzung von Plattennamen (Volume Names)
1.2.2 Bilden von Directories
1.2.3 Anlegen von Dateien

1.3 Zuweisung von Synonymen (AS Kommando)
1.3.1 Format des AS Kommandos
1.3.2 Benutzerantworten beim AS Kommando
1.3.3 Beispiel zum AS Kommando

1.4 Anzeigen der Synonymwerte (LS Kommando)
1.4.1 Format des LS Kommandos
1.4.2 Benutzerantworten beim LS Kommando
1.4.3 Beispiel zum LS Kommando

1.5 Modifizieren von Synonymen (MS Kommando)
1.5.1 Format des MS Kommandos
1.5.2 Benutzerantworten beim MS Kommando
1.5.3 Beispiel zum MS Kommando

1.6 Löschen aller Synonyme (DAS Kommando)
1.6.1 Format des DAS Kommandos
1.6.2 Benutzerantworten beim DAS Kommando
1.6.3 Beispiel zum DAS Kommando

1.7 Zuordnung alternativer Namen (ALIASES) zu Namen von Directories oder Dateien (AA Kommando)
1.7.1 Format des AA Kommandos
1.7.2 Benutzerantworten beim AA Kommando
1.7.3 Beispiel zum AA Kommando

1.8 Löschen alternativer Namen (DA Kommando)
1.8.1 Format des DA Kommandos
1.8.2 Benutzerantworten beim DA Kommando
1.8.3 Beispiel zum DA Kommando

1.9 Ändern von Dateinamen (MFN Kommando)
1.9.1 Format des MFN Kommandos
1.9.2 Benutzerantworten beim MFN Kommando
1.9.3 Beispiel zum MFN Kommando

INHALTSVERZEICHNIS (Fortsetzung)

1.10 Änderung des Schutzes einer Datei (MFP Kommando)
1.10.1 Format des MFP Kommandos
1.10.2 Benutzerantworten beim MFP Kommando
1.10.3 Beispiel zum MFP Kommando

1.11 Zuweisen von logischen Namen (ALN Kommando)
1.11.1 Format des ALN Kommandos
1.11.2 Benutzerantworten beim ALN Kommando
1.11.3 Beispiel zum ALN Kommando

1.12 Löschen von logischen Namen (RLN Kommando)
1.12.1 Format des RLN Kommandos
1.12.2 Benutzerantworten beim RLN Kommando
1.12.3 Beispiel zum RLN Kommando

2 AUSGABE VON DATEIEN UND DIRECTORIES AUF BILDSCHIRM ODER DRUCKER

2.1 Allgemeines

2.2 Inhalte von Directories auflisten (LD Kommando)
2.2.1 Format des LD Kommandos
2.2.2 Benutzerantworten beim LD Kommando
2.2.3 Beispiel zum LD Kommando

2.3 Platteninhalte auflisten (MD Kommando)
2.3.1 MD Kommando: Wahlmöglichkeit SHORT FORM
2.3.2 MD Kommando: Wahlmöglichkeit TOP LEVEL ONLY
2.3.3 MD Kommando: Wahlmöglichkeit DIRECTORY NODES ONLY
2.3.4 Format des MD Kommandos
2.3.5 Benutzerantworten beim MD Kommando
2.3.6 Beispiel zum MD Kommando

2.4 Auflistung der Schlüsselbeschreibungen von Key Index Dateien (MKF Kommando)
2.4.1 Format des MKF Kommandos
2.4.2 Benutzerantworten beim MKF Kommando
2.4.3 Beispiel zum MKF Kommando

2.5 Auflisten des Inhalts einer Programmdatei (MPF Kommando)
2.5.1 Format des MPF Kommandos
2.5.2 Benutzerantworten beim MPF Kommando
2.5.3 Beispiel zum MPF Kommando

2.6 Auflisten der Inhalte logischer Sätze (LLR Kommando)
2.6.1 Format des LLR Kommandos
2.6.2 Benutzerantworten beim LLR Kommando
2.6.3 Beispiel zum LLR Kommando

2.7 Ausgabe von Dateiinhalten (SF Kommando)
2.7.1 Format des SF Kommandos
2.7.2 Benutzerantworten beim SF Kommando
2.7.3 Beispiel zum SF Kommando

2.8 Ausgeben von Dateiinhalten auf einem Drucker - der TAXO Spooler (PF Kommando)
2.8.1 Erstellen von Druckaufträgen
2.8.1.1 Format des PF Kommandos
2.8.1.2 Benutzerantworten beim PF Kommando
2.8.1.3 Beispiel zum PF Kommando

INHALTSVERZEICHNIS (Fortsetzung)

2.8.2 Druckunterbrechung (HO Kommando)
2.8.2.1 Format des HO Kommandos
2.8.2.2 Benutzerantworten beim HO Kommando
2.8.2.3 Beispiel zum HO Kommando

2.8.3 Fortsetzen der Ausgabe (RO Kommando)
2.8.3.1 Format des RO Kommandos
2.8.3.2 Benutzerantworten beim RO Kommando
2.8.3.3 Beispiel zum RO Kommando

2.8.4 Beenden einer Ausgabe (KO Kommando)
2.8.4.1 Format des KO Kommandos
2.8.4.2 Benutzerantworten beim KO Kommando
2.8.4.3 Beispiel zum KO Kommando

2.8.5 Anzeigen des Spooler Status (SOS Kommando)
2.8.5.1 Format des SOS Kommandos
2.8.5.2 Benutzerantworten beim SOS Kommando
2.8.5.3 Beispiel zum SOS Kommando

2.8.6 Formularwechsel (MSD Kommando)
2.8.6.1 Format des MSD Kommandos
2.8.6.2 Benutzerantworten beim MSD Kommando
2.8.6.3 Dialogmeldungen des MSD Kommandos
2.8.6.4 Beispiel zum MSD Kommando

2.8.7 Erstellen eines Ausschnitts aus einer Druckdatei für Druckwiederholung (BPPF Kommando)
2.8.7.1 Format des BPPF Kommandos
2.8.7.2 Benutzerantworten beim BPPF Kommando
2.8.7.3 Beispiel zum BPPF Kommando

2.8.8 History File des Spooler Systems (S$PF Kommando)
2.8.8.1 Anzeigen des Spooler History Files
2.8.8.1.1 Benutzerantworten beim S$PF Kommando
2.8.8.1.2 Beispiel zum S$PF Kommando

2.8.8.2 Löschen von im History File enthaltenen Druckdateien (D$PF Kommando)
2.8.8.2.1 Benutzerantworten beim D$PF Kommando
2.8.8.2.2 Beispiel zum D$PF Kommando

2.8.9 Einstellen der Druckbreite (LPWIDTH Kommando)
2.8.9.1 Format des LPWIDTH Kommandos
2.8.9.2 Benutzerantworten beim LPWIDTH-Kommando
2.8.9.3 Beispiel zum LPWIDTH Kommando

2.8.10 International PRINT FILE (IPF Kommando)
2.8.10.1 Format des IPF Kommandos
2.8.10.2 Benutzerantworten beim IPF Kommando
2.8.10.3 Beispiel zum IPF Kommando
2.8.10.4 Fehlermeldungen vom IPF Kommando

2.8.11 Erzeugen eines Druckauftrags aus einem Anwendungsprogramm

2.8.12 Formularsteuerdateien
2.8.12.1 Aufbau von Formularsteuerdateien
2.8.12.1.1 Pfadnamen für Formularsteuerdateien
2.8.12.1.2 Format der Formularsteuerdateien

2.8.13 Ausgabe und Behandlung von Fehlermeldungen des Spoolers

INHALTSVERZEICHNIS (Fortsetzung)

3 KOPIEREN VON DATEIEN UND DIRECTORIES

3.1 Allgemeines

3.2 Kopieren und Verketten von Dateien (CC Kommando)
3.2.1 Format des CC Kommandos
3.2.2 Benutzerantworten beim CC Kommando

3.3 Anhängen einer oder mehrerer Dateien an eine andere Datei (AF Kommando)
3.3.1 Format des AF Kommandos
3.3.2 Benutzerantworten beim AF Kommando
3.3.3 Beispiel zum AF Kommando

3.4 Modifizieren der Aufzeichnung bei Key Index Dateien (MKL Kommando)
3.4.1 Format des MKL Kommandos
3.4.2 Benutzerantworten beim MKL Kommando
3.4.3 Beispiel zum MKL Kommando

3.5 Übertragen sequentieller Dateien in Key Index Dateien (CSK Kommando)
3.5.1 Format des CSK Kommandos
3.5.2 Benutzerantworten beim CSK Kommando
3.5.3 Beispiel zum CSK Kommando

3.6 Übertragung von Key Index Dateien nach sequentiellen Dateien (CKS Kommando)
3.6.1 Format des CKS Kommandos
3.6.2 Benutzerantworten beim CKS Kommando
3.6.3 Beispiel zum CKS Kommando

3.7 Copying Key Indexed Files to Sequential Files Randomly (CKSR Kommando)
3.7.1 Format des CKSR Kommandos
3.7.2 Benutzerantworten beim CKSR Kommando
3.7.3 Beispiel zum CKSR Kommando
3.7.4 CKSR Listdatei Meldungen

3.8 Copy Program Image (CPI Kommando)
3.8.1 Format des CPI Kommandos
3.8.2 Benutzerantworten beim CPI Kommando
3.8.3 Beispiel zum CPI Kommando

3.9 Directory Kommandos (CD Kommando)
3.9.1 Kopieren von Directories (CD Kommando)
3.9.1.1 Format des CD Kommandos
3.9.1.2 Benutzerantworten beim CD Kommando
3.9.1.3 Beispiel zum CD Kommando

3.9.2 Verifizieren einer Kopie eines Directory (VC Kommando)
3.9.2.1 Format des VC Kommandos
3.9.2.2 Benutzerantworten beim VC Kommando
3.9.2.3 Beispiel zum VC Kommando

3.9.3 Anfertigen von Sicherungskopien von Directories (BD Kommando)
3.9.3.1 Format des BD Kommandos
3.9.3.2 Benutzerantworten beim BD Kommando
3.9.3.3 Beispiel zum BD Kommando

INHALTSVERZEICHNIS (Fortsetzung)

3.9.4 Wiederherstellen eines Directory (RD Kommando)
3.9.4.1 Format des RD Kommandos
3.9.4.2 Benutzerantworten beim RD Kommando
3.9.4.3 Beispiel zum RD Kommando

3.9.5 Prüfen einer Wiederherstellung oder einer Sicherung (VB Kommando)
3.9.5.1 Format des VB Kommandos
3.9.5.2 Benutzerantworten beim VB Kommando
3.9.5.3 Beispiel zum VB Kommando

3.10 Kontrolldateien

4 PROGRAMMAUSFÜHRUNG UND KONTROLLE

4.1 Allgemeines

4.2 Ausführen nichtinteraktiver Tasks (XT Kommando)
4.2.1 Format des XT Kommandos
4.2.2 Benutzerantworten beim XT Kommando
4.2.3 Beispiel zum XT Kommando

4.3 Ausführen interaktiver Tasks (XTS Kommando)
4.3.1 Format des XTS Kommandos
4.3.2 Benutzerantworten beim XTS Kommando
4.3.3 Beispiel zum XTS Kommando

4.4 Ausführen und Anhalten einer Task (EXECUTE AND HALT TASK - XHT)

4.5 Anzeigen des Status von Tasks (STS Kommando)
4.5.1 Format des STS Kommandos
4.5.2 Benutzerantworten beim STS Kommando
4.5.3 Beispiel zum STS Kommando

4.6 Beenden von Tasks (KT Kommando)
4.6.1 Format des KT Kommandos
4.6.2 Benutzerantworten beim KT Kommando
4.6.3 Beispiel zum KT Kommando

4.7 Programmabbruch durch Tastendruck (Hard Break Sequence)

INHALTSVERZEICHNIS (Fortsetzung)

5 KONTROLLE DES TERMINALZUGRIFFS

5.1 Allgemeines

5.2 Verändern des Terminalstatus (MTS Kommando)
5.2.1 Format des MTS Kommandos
5.2.2 Benutzerantworten beim MTS Kommando
5.2.3 Beispiel zum MTS Kommando

5.3 Auflisten des Terminalstatus (LTS Kommando)
5.3.1 Format des LTS Kommandos
5.3.2 Benutzerantworten beim LTS Kommando
5.3.3 Beispiel zum LTS Kommando

5.4 Kontrolle der Benutzeridentifikation (AUI Kommando)
5.4.1 Zuweisung von Benutzeridentifikationen (AUI Kommando)
5.4.1.1 Format des AUI Kommandos
5.4.1.2 Benutzerantworten beim AUI Kommando
5.4.1.3 Beispiel zum AUI Kommando

5.4.2 Verändern von Benutzeridentifikationen (MUI Kommando)
5.4.2.1 Format des MUI Kommandos
5.4.2.2 Beispiel zum MUI Kommando

5.4.3 Löschen von Benutzeridentifikationen (DUI Kommando)
5.4.3.1 Format des DUI Kommandos
5.4.3.2 Benutzerantworten beim DUI Kommando
5.4.3.3 Beispiel zum DUI Kommando

5.4.4 Auflisten von Benutzeridentifikationen (LUI Kommando)
5.4.4.1 Format des LUI Kommandos
5.4.4.2 Benutzerantworten beim LUI Kommando
5.4.4.3 Beispiel zum LUI Kommando

INHALTSVERZEICHNIS (Fortsetzung)

6 INSTALLIEREN, ZUWEISEN VON ERSATZSTELLEN, ANMELDEN, ABMELDEN UND ANZEIGEN DES STATUS VON PLATTEN

6.1 Allgemeines

6.2 Initialisieren neuer Platten (INV Kommando)
6.2.1 Format des INV Kommandos
6.2.2 Benutzerantworten beim INV Kommando
6.2.3 Beispiel zum INV Kommando

6.3 Kommandos zum Zuweisen von Ersatzstellen für defekte Stellen auf einer Platte

6.3.1 Allgemeines

6.3.2 Zuweisen von Ersatzstellen (ABL Kommando)
6.3.2.1 Format des ABL Kommandos
6.3.2.2 Benutzerantworten beim ABL Kommando
6.3.2.3 Beispiel zum ABL Kommando

6.3.3 Zurücksetzen der zugewiesenen Ersatzstellen (RBL Kommando)
6.3.3.1 Format des RBL Kommandos
6.3.3.2 Benutzerantworten beim RBL Kommando
6.3.3.3 Beispiel zum RBL Kommando

6.3.4 Anzeigen der Ersatzstellenliste einer Platte (SBL Kommando)
6.3.4.1 Format des SBL Kommandos
6.3.4.2 Benutzerantworten beim SBL Kommando
6.3.4.3 Beispiel zum SBL Kommando

6.3.5 IDS Kommando
6.3.5.1 Format des IDS Kommandos
6.3.5.2 Benutzerantworten beim IDS Kommando
6.3.5.3 Beispiel zum IDS Kommando

6.4 Installieren von Platten (IV Kommando)
6.4.1 Format des IV Kommandos
6.4.2 Benutzerantworten beim IV Kommando
6.4.3 Beispiel zum IV Kommando

6.5 Abmelden von Platten (UV Kommando)
6.5.1 Format des UV Kommandos
6.5.2 Benutzerantworten beim UV Kommando
6.5.3 Beispiel zum UV Kommando

6.6 Anzeigen des Plattenstatus (SVS Kommando)
6.6.1 Format des SVS Kommandos
6.6.2 Benutzerantworten beim SVS Kommando
6.6.3 Beispiel zum SVS Kommando

INHALTSVERZEICHNIS (Fortsetzung)

7 ANLEGEN UND LÖSCHEN VON DATEIEN UND DIRECTORIES
7.1 Allgemeines
7.2 Anlegen einer beliebigen Datei (CF Kommando)
7.2.1 Format des CF Kommandos
7.2.2 Benutzerantworten beim CF Kommando
7.2.3 Beispiele zum CF Kommando
7.3 Anlegen von Directory-Dateien (CFDIR Kommando)
7.3.1 Format des CFDIR Kommandos
7.3.2 Benutzerantworten beim CFDIR Kommando
7.3.3 Beispiel zum CFDIR Kommando
7.4 Anlegen von Key Index Dateien (CFKEY Kommando)
7.4.1 Format des CFKEY Kommandos
7.4.2 Benutzerantworten beim CFKEY Kommando
7.4.3 Beispiel zum CFKEY Kommando
7.4.4 CFKEY im BATCH MODUS
7.4.5 Neue Key Indexed File Organisation
7.4.5.1 Formeln zur Berechnung der Größe von Key Indexed Dateien
7.4.5.2 Prelogging Area
7.4.5.3 B-Tree Node Area
7.4.5.4 Data Block Area
7.4.5.5 Gesamtgröße
7.4.5.6 Beispiel 1
7.4.5.7 Beispiel 2
7.5 Anlegen einer Relativ Record Datei (CFREL Kommando)
7.5.1 Format des CFREL Kommandos
7.5.2 Benutzerantworten beim CFREL Kommando
7.5.3 Beispiel zum CFREL Kommando
7.6 Anlegen sequentieller Dateien (CFSEQ Kommando)
7.6.1 Format des CFSEQ Kommandos
7.6.2 Benutzerantworten beim CFSEQ Kommando
7.6.3 Beispiel zum CFSEQ Kommando
7.7 Erzeugen von Programm Dateien (CFPRO Kommando)
7.7.1 Format des CFPRO Kommandos
7.7.2 Benutzerantworten beim CFPRO Kommando
7.7.3 Beispiel zum CFPRO Kommando
7.8 Erzeugen von Image Dateien (CFIMG Kommando)
7.8.1 Format des CFIMG Kommandos
7.8.2 Benutzerantworten beim CFIMG Kommando
7.8.3 Beispiel zum CFIMG Kommando
7.9 Löschen von Directories (DD Kommando)
7.9.1 Format des DD Kommandos
7.9.2 Benutzerantworten beim DD Kommando
7.9.3 Beispiel zum DD Kommando
7.10 Löschen von Dateien (DF Kommando)
7.10.1 Format des DF Kommandos
7.10.2 Benutzerantworten beim DF Kommando
7.10.3 Beispiel zum DF Kommando

INHALTSVERZEICHNIS (Fortsetzung)

8 BATCH MODUS DES SYSTEM COMMAND INTERPRETERS

8.1 Allgemeines

8.2 Ausführen des SCI im Batch Modus (XB Kommando)
8.2.1 Format des XB Kommandos
8.2.2 Benutzerantworten beim XB Kommando
8.2.3 Beispiel zum XB Kommando

8.3 Zustandsanzeige von Tasks im Hintergrund (SBS Kommando)
8.3.1 Beispiel zum SBS Kommando
8.3.2 KEYWORD STATE SYNONYM (Schlüsselwort)

8.4 Beenden von Tasks im Hintergrund (KBT Kommando)
8.4.1 Format des KBT Kommandos
8.4.2 Benutzerantworten beim KBT Kommando
8.4.3 Beispiel zum KBT Kommando

8.5 Format des Batch Kommandos

8.6 Format des BATCH COMMAND STREAMS
8.6.1 Format des BATCH Kommandos
8.6.2 Format des EBATCH Kommandos

8.7. Warten auf die Beendigung auf Backgroundaktivitäten (WAIT Kommando)

8.8 Zählen von Fehlern in Batchstreams (EC Kommando)

8.9 Expert Mode

INHALTSVERZEICHNIS (Fortsetzung)

9 ZUWEISUNG UND KONTROLLE EINER LOGISCHEN EINHEIT

9.1 Allgemeines

9.2 Zuweisung einer globalen logischen Einheit (AGL Kommando)
9.2.1 Format des AGL Kommandos
9.2.2 Benutzerantwort beim AGL Kommando
9.2.3 Beispiel zum AGL Kommando

9.3 Aufheben einer globalen LUNO (RGL Kommando)
9.3.1 Format des RGL Kommandos
9.3.2 Benutzerantwort beim RGL Kommando
9.3.3 Beispiel zum RGL Kommando

9.4 Zuweisung einer Terminal-Local logischen Einheit (AL Kommando)
9.4.1 Format des AL Kommandos
9.4.2 Benutzerantworten beim AL-Kommando
9.4.3 Beispiel zum AL Kommando

9.5 Aufheben einer Terminal-Local LUNO (RL Kommando)
9.5.1 Format des RL Kommandos
9.5.2 Benutzerantworten beim RL Kommando
9.5.3 Beispiel zum RL Kommando

9.6 Aufheben aller Terminal-Local LUNO Zuweisungen (RAL Kommando)
9.6.1 Format des RAL Kommandos
9.6.2 Benutzerantworten beim RAL Kommando
9.6.3 Beispiel zum RAL Kommando

9.7 Vorwärtspositionieren logischer Einheiten (FL Kommando)
9.7.1 Format des FL Kommandos
9.7.2 Benutzerantworten beim FL Kommando
9.7.3 Beispiel zum FL Kommando

9.8 Rücksetzen einer logischen Einheit (BL Kommando)
9.8.1 Format des BL Kommandos
9.8.2 Benutzerantworten beim BL Kommando
9.8.3 Beispiel zum BL Kommando

9.9 Zurückspulen logischer Einheiten (RWL Kommando)
9.9.1 Format des RWL Kommandos
9.9.2 Benutzerantworten beim RWL Kommando
9.9.3 Beispiel zum RWL Kommando

9.10 Anzeigen des I/O Status einer logischen Einheit (SIS Kommando)
9.10.1 Format des SIS Kommandos
9.10.2 Benutzerantworten beim SIS Kommando
9.10.3 Beispiel zum SIS Kommando

9.11 Modifizieren des Schutzes einer globalen LUNO gegen Freigeben (MLP Kommando)
9.11.1 Format des MLP Kommandos
9.11.2 Benutzerantworten beim MLP Kommando
9.11.3 Beispiel zum MLP Kommando

9.12 WRITE END-OF-FILE TO LUNO (WEOF Kommando)
9.12.1 Format des WEOF Kommandos
9.12.2 Benutzerantworten beim WEOF Kommando
9.12.3 Beispiel zum WEOF Kommando

INHALTSVERZEICHNIS (Fortsetzung)

10 SONSTIGE KOMMANDOS

10.1 Allgemeines

10.2 Anzeige von Systemdatum und Systemzeit (SDT Kommando)
10.2.1 Format des SDT Kommandos
10.2.2 Benutzerantworten beim SDT Kommando
10.2.3 Beispiel zum SDT Kommando

10.3 Senden einer Nachricht zu einem Terminal (CM Kommando)
10.3.1 Format des CM Kommandos
10.3.2 Benutzerantworten beim CM Kommando
10.3.3 Beispiel zum CM Kommando

10.3.4 SCI Anschluß (CSLM Kommando)
10.3.4.1 Format des CSLM Kommandos
10.3.4.2 Benutzerantworten beim CSLM Kommando
10.3.4.3 Beispiele zum CSLM Kommando

10.4 Anzeigen von Terminalinformationen (STI Kommando)
10.4.1 Format des STI Kommandos
10.4.2 Benutzerantworten beim STI Kommando
10.4.3 Beispiel zum STI Kommando

10.5 Anzeige des Speicherzustandes (SMS Kommando)
10.5.1 Format des SMS Kommandos
10.5.2 Benutzerantworten beim SMS Kommando
10.5.3 Beispiel zum SMS Kommando

10.6 Ausgabe der Systemgeräte - Konfiguration (LDC Kommando)
10.6.1 Format des LDC Kommandos
10.6.2 Benutzerantworten beim LDC Kommando
10.6.3 Beispiel zum LDC Kommando

10.6.4 Anzeige des Hauptspeichers am Bildschirm (SMM Kommando)
10.6.4.1 Format des SMM Kommandos
10.6.4.2 Benutzerantworten beim SMM Kommando
10.6.4.3 Beispiel zum SMM Kommando

10.7 Allgemeine Kommandos
10.7.1 Die Kommandos SLG, SPDEV,QPDEV
10.7.2 Eintragen einer Meldung in das System Logbuch (SLG Kommando)
10.7.2.1 Format des SLG Kommandos
10.7.2.2 Benutzerantworten beim SLG Kommando
10.7.2.3 Beispiel zum SLG Kommando

10.7.3 Zugriff auf die allgemeinen Systemkommandos (SPDEV Kommando)
10.7.3.1 SPDEV Kommando (Start Program Development)
10.7.3.2 QPDEV Kommando (Quit Program Development)

10.7.4 SHOW DTS LOG (SDTSL Kommando)
10.7.4.1 Format des SDTSL Kommandos
10.7.4.2 Benutzerantworten beim SDTSL Kommando
10.7.4.3 Beispiel zum SDTSL Kommando

10.7.5 Clear Secret Synonyms (Q$SYN Kommando)

10.7.6 COBOL Anschluß

INHALTSVERZEICHNIS (Fortsetzung)

10.7.7 Unterstützung länderspezifischer Zeichenvorräte (Country Code)
10.7.7.1 Allgemeines
10.7.7.2 Anzeigen des im System eingestellten Country Codes (SCC Kommando)
10.7.7.2.1 Format des SCC Kommandos
10.7.7.2.2 Benutzerantworten beim SCC Kommando
10.7.7.2.3 Beispiel zum SCC Kommando
10.7.7.3 Ändern des im System eingestellten Country Codes (MCC Kommando)
10.7.7.3.1 Format des MCC Kommandos
10.7.7.3.2 Benutzerantworten beim MCC Kommando
10.7.7.3.3 Beispiel zum MCC Kommando
10.7.8 Anzeigen von Fehlermeldungen am Benutzerterminal

11 GEBRAUCH DES LINK EDITORS
11.1 Merkmale
11.2 Arbeitsweise des Link Editors

12 INSTALLIEREN, ANZEIGEN, VERÄNDERN UND LÖSCHEN VON PROGRAMMEN IN PROGRAMMDATEIEN
12.1 Einführung
12.2 Aufbau von TAXO Programmdateien
12.3 Merkmale von Modulen eines Programms
12.4 Installieren von Modulen eines Programms
12.4.1 Installieren einer Task (IT Kommando)
12.4.1.1 Format des IT (Install Task) Kommandos
12.4.1.2 Benutzerantworten beim IT (Install Task) Kommando
12.4.1.3 Beispiel zum IT Kommando
12.4.2 Installieren einer Echtzeit Task (IRT Kommando)
12.4.2.1 Format des IRT (Install Realtime Task) Kommandos
12.4.2.2 Benutzerantworten beim IRT (Install Realtime Task) Kommando
12.4.2.3 Beispiel zum IRT Kommando
12.4.3 Installieren einer Prozedur (IP Kommando)
12.4.3.1 Format des IP (Install Procedure) Kommandos
12.4.3.2 Benutzerantworten beim IP (Install Procedure) Kommando
12.4.3.3 Beispiel zum IP Kommando
12.4.4 Installieren eines Overlays (IO Kommando)
12.4.4.1 Format des IO (Install Overlay) Kommandos
12.4.4.2 Benutzerantworten beim IO (Install Overlay) Kommando
12.4.4.3 Beispiel zum IO Kommando
12.5 Löschen von Modulen eines Programms
12.5.1 Löschen einer Task (DT Kommando)
12.5.1.1 Format des DT (Delete Task) Kommandos
12.5.1.2 Benutzerantworten beim DT (Delete Task) Kommando
12.5.1.3 Beispiel zum DT Kommando

INHALTSVERZEICHNIS (Fortsetzung)

12.5.2 Löschen einer Prozedur (DP Kommando)
12.5.2.1 Format des DP (Delete Prozedur) Kommandos
12.5.2.2 Benutzerantworten beim DP (Delete Prozedur) Kommando
12.5.2.3 Beispiel zum DP Kommando

12.5.3 Löschen eines Overlays (DO Kommando)
12.5.3.1 Format des DO (Delete Overlay) Kommandos
12.5.3.2 Benutzerantworten beim DO (Delete Overlay) Kommando
12.5.3.3 Beispiel zum DO Kommando

12.6 Ändern der Moduleinträge in Programmdateien

12.6.1 Ändern von Taskeinträgen (MTE Kommando)
12.6.1.1 Interaktive Kommandoausführung
12.6.1.2 Batch Kommandoausführung
12.6.1.3 Beispiel zum MTE (Modify Task Entry) Kommando

12.6.2 Ändern von Prozedureinträgen (MPE Kommando)
12.6.2.1 Interaktive Kommandoausführung
12.6.2.2 Batch Kommandoausführung
12.6.2.3 Beispiel zum MPE Kommando

12.6.3 Ändern von Overlayeinträgen (MOE Kommando)
12.6.3.1 Interaktive Kommandoausführung
12.6.3.2 Batch Kommandoausführung
12.6.3.3 Beispiel zum MOE Kommando

12.7 Anzeigen des Programmcodes eines Moduls (SPI Kommando)
12.7.1 Format des SPI Kommandos
12.7.2 Benutzerantworten beim SPI Kommando
12.7.3 Beispiel zum SPI Kommando

12.8 Ändern des Programmcodes eines Moduls (MPI Kommando)
12.8.1 Format des MPI Kommandos
12.8.2 Benutzerantworten beim MPI Kommando
12.8.3 Beispiel zum MPI Kommando

12.9 Beispiele zu den Installations-, Änderungs- und Löschkommandos für Programme

13 KOMMANDOS FÜR DIE DATENSICHERUNG

13.1 Allgemeines

13.2 Kopieren eines sequentiellen Mediums (CSM Kommando)
13.2.1 Format des CSM Kommandos
13.2.2 Benutzerantworten beim CSM Kommando
13.2.3 Beispiel zum CSM Kommando

13.3 Kopieren von Platten (DCOPY Kommando)

13.3.1 Durchführungsprozedur für DCOPY
13.3.1.1 DCOPY für den Standardkopiervorgang (Copy Mode)
13.3.1.2 DCOPY für den nicht standardmäßigen Kopiervorgang (Salvage Mode)

13.3.2 DCOPY Fehlermeldungen
13.3.3 Beispiele für DCOPY Operationen
13.3.4 Reorganisieren von Platten mit Reorganisation (CVD Kommando)

INHALTSVERZEICHNIS (Fortsetzung)

14 INITIALISIEREN UND REINITIALISIEREN VON BETRIEBSSYSTEM-FUNKTIONEN

14.1 Allgemeines

14.2 System-Warmstart Prozedur (IS Kommando)
14.2.1 Format des IS Kommandos
14.2.2 Benutzerantworten beim IS Kommando
14.2.3 Beispiel zum IS Kommando

14.3 Reinitialisieren von Systemdatum und Systemzeit (IDT Kommando)
14.3.1 Format des IDT Kommandos
14.3.2 Benutzerantworten beim IDT Kommando
14.3.3 Beispiel zum IDT Kommando

14.4 Retten des System Log File

14.5 Reinitialisieren des System Log (ISL Kommando)
14.5.1 Format des ISL Kommandos
14.5.2 Benutzerantworten beim ISL Kommando
14.5.3 Beispiel zum ISL Kommando
14.5.4 Standardwerte für das ISL Kommando beim Aufruf durch IS

15 KOMMANDOS ZUR PFLEGE DES BETRIEBSSYSTEMS

15.1 System Dateien und Directories

15.2 Modifizieren von Einträgen im Systembereich einer Platte (MVI Kommando)
15.2.1 Format des MVI Kommandos
15.2.2 Beispiel zum MVI Kommando

15.3 Anlegen von System Dateien (CSF Kommando)
15.3.1 Format des CSF Kommandos
15.3.2 Benutzerantworten beim CSF Kommando
15.3.3 Beispiel zum CSF Kommando

15.4 Installieren eines System Overlay (ISO Kommando)
15.4.1 Format des ISO Kommandos
15.4.2 Benutzerantworten beim ISO Kommando

16 DISKETTENDIENSTPROGRAMME

16.1 Allgemeines

16.2 TAXO - TASO Datei Konvertierungsprogramme
16.2.1 Allgemeines
16.2.2 Dienstprogramm zum Konvertieren von Dateien (FCU Kommando)
16.2.2.1 Format des FCU Kommandos
16.2.2.2 Benutzerantworten beim FCU Kommando
16.2.2.3 Die Anweisungen für das FCU Kommando
16.2.2.4 Beschreibung der Kommandoanweisungen
16.2.2.5 Sonstige Fehlermeldungen
16.2.2.6 SCI-Kommandos zum Aufruf des FCU Programms
16.2.2.7 Beispiel zum FCU Kommando
16.2.2.8 Zusammenfassung der Fehlermeldungen

INHALTSVERZEICHNIS (Fortsetzung)

16.2.3 Dienstprogramm zum Initialisieren von TA 1600/20 / TA 1600/10 Disketten (INITASO Kommando)
16.2.3.1 Format des INITASO Kommandos
16.2.3.2 Benutzerantworten beim INITASO Kommando
16.2.3.3 Beispiel zum INITASO Kommando

16.2.4 Auflisten des Inhalts einer TASO-Diskette (TASOMD Kommando)
16.2.4.1 Format des TASOMD Kommandos
16.2.4.2 Benutzerantworten beim TASOMD Kommando
16.2.4.3 Beispiel zum TASOMD Kommando

16.3 Dienstprogramme für IBM kompatible Disketten
16.3.1 Allgemeines
16.3.2 Dienstprogramm zum Übertragen von Daten zwischen TAXO Dateien und IBM Datasets (COPYIBM Kommando)
16.3.2.1 Format des COPYIBM Kommandos
16.3.2.2 Benutzerantworten beim COPYIBM Kommando
16.3.2.3 Beispiel zum COPYIBM Kommando
16.3.2.5 Meldungen zur Benutzerführung

16.3.3 Dienstprogramm zum Initialisieren einer Diskette im Standard IBM Format (INITIBM Kommando)
16.3.3.1 Format des INITIBM Kommandos
16.3.3.2 Benutzerantworten beim INITIBM Kommando
16.3.3.3 Beispiel zum INITIBM Kommando

16.3.4 Dienstprogramm zum Auflisten der Struktur einer im IBM Format formatiesierten Diskette (IBMMD Kommando)
16.3.4.1 Format des IBMMD Komandos
16.3.4.2 Benutzerantworten beim IBMMD Kommando
16.3.4.3 Beispiel zum IBMMD Kommando

16.4 Formatumschaltung des Disketten Controllers (SDF Kommando)
16.4.1 Einstellen des Diskettenformats (SDF Kommando)
16.4.1.1 Format des SDF Kommandos
16.4.1.2 Benutzerantworten beim SDF Kommando
16.4.1.3 Beispiel zum SDF Kommando

16.4.2 Zurückstellen des Diskettenformats auf TAXO Standard (RDF Kommando)
16.4.2.1 Format des RDF Kommandos
16.4.2.2 Benutzerantworten beim RDF Kommando
16.4.2.3 Beispiel zum RDF Kommando

16.4.3 Auflisten des eingestellten Diskettenformats (LDF Kommando)
16.4.3.1 Format des LDF Kommandos
16.4.3.2 Benutzerantworten beim LDF Kommando
16.4.3.3 Beschreibung der Ausgabeliste des LDF Kommandos
16.4.3.4 Beispiel zum LDF Kommando

16.5 Physikalische Diskettenkommandos

16.5.1 Auflisten von Sektoren von Disketten (DTS Kommando)
16.5.1.1 Format des DTS Kommandos
16.5.1.2 Benutzerantworten beim DTS Kommando
16.5.1.3 Beispiel zum DTS Kommando

16.5.2 Modifizieren eines Sektors auf einer Diskette (DMTS Kommando)
16.5.2.1 Format des DMTS Kommandos
16.5.2.2 Benutzerantworten beim DMTS Kommando
16.5.2.3 Beispiel zum DMTS Kommando

INHALTSVERZEICHNIS (Fortsetzung)

17 DIENSTPROGRAMME FÜR DIE FEHLERSUCHE UND FEHLERBEHEBUNG AUF DATENTRÄGERN

17.1 Allgemeines

17.2 Dienstprogramme zum Prüfen von Magnetplatten
17.2.1 Allgemeines
17.2.2 Überprüfen der Konsistenz von Platten (CKD Kommando)
17.2.2.1 Format des CKD Kommandos
17.2.2.2 Benutzerantworten beim CKD Kommando
17.2.2.3 Durchführungsprozedur für das CKD Kommando
17.2.2.4 Analyse des Protokolls des CKD Kommandos
17.2.2.5 Beispiele zum CKD Kommando

17.3 Dienstprogramme zum Bestimmen der Zuordnung von physikalischen Plattenadressen zu logischen Dateien
17.3.1 Auswertung der Logbucheinträge bei Plattenfehlern
17.3.2 Das FFADU Kommando (Find File for ADU)
17.3.2.1 Format des FFADU Kommandos
17.3.2.2 Benutzerantworten beim FFADU Kommando
17.3.2.3 Beispiel zum FFADU Kommando

17.3.3 Das FFTS Kommando (Find File for Track Sector)
17.3.3.1 Format des FFTS Kommandos
17.3.3.2 Benutzerantworten beim FFTS Kommando
17.3.3.3 Beispiel zum FFTS Kommando

17.3.4 Das FFCHS Kommando (Find File for Cylinder, Head and Sector)
17.3.4.1 Format des FFCHS Kommandos
17.3.4.2 Benutzerantworten beim FFCHS Kommando
17.3.4.3 Beispiel zum FFCHS Kommando

17.3.5 Das LADU Kommando (List ADUs)
17.3.5.1 Format des LADU Kommandos
17.3.5.2 Benutzerantworten beim LADU Kommando
17.3.5.3 Beispiel zum LADU Kommando

17.3.6 Fehlermeldungen bei den Kommandos FFADU, FFTS, FFCHS und LADU

17.3.7 RECOVER DISK (RCD Kommando)
17.3.7.1 Format des RCD Kommandos
17.3.7.2 Benutzerantworten beim RCD Kommando
17.3.7.3 Beispiele zum RCD Kommando

17.4 Physikalisches Anzeigen von Dateiinhalten (SRF Kommando)
17.4.1 Format des SRF Kommandos
17.4.2 Benutzerantworten beim SRF Kommando
17.4.3 Beispiele zum SRF Kommando

17.5 Ändern von Dateiinhalten (MRF Kommando)
17.5.1 Format des MRF Kommandos
17.5.2 Benutzerantworten beim MRF Kommando
17.5.3 Beispiel zum MRF Kommando

17.6 Kommandos zum Anzeigen von Platteninhalten (SAD und SADU Kommando)
17.6.1 Format des SAD und SADU Kommandos
17.6.2 Benutzerantworten beim SAD und SADU Kommando
17.6.3 Beispiele zum SAD und SADU Kommando

INHALTSVERZEICHNIS (Fortsetzung)

17.7 Kommandos zum Ändern von Platteninhalten (MAD und MADU Kommando)
17.7.1 Format des MAD und MADU Kommandos
17.7.2 Benutzerantworten beim MAD und MADU Kommando
17.7.3 Beispiele zum MAD und MADU Kommando

17.8 Ergänzen eines fehlenden "End of File" - Kennzeichens für eine sequentielle Datei (REOF Kommando)
17.8.1 Format des REOF Kommandos
17.8.2 Benutzerantworten beim REOF Kommando
17.8.3 Beispiel zum REOF Kommando

17.9 Modify Device State (MDS Kommando)
17.9.1 Format des MDS Kommandos
17.9.2 Benutzerantworten beim MDS Kommando

A TASK STATUS CODES

A.1 Überblick
A.2 Programmbibliothek S$PROGA
A.3 Programmbibliothek TA$PROG

B GLOBALE LUNOS UND ZULÄSSIGE GERÄTENAMEN

B.1 Überblick

C ALPHABETISCHES VERZEICHNIS DER SCI KOMMANDOS

D KEY INDEXED DATEI ORGANISATION

D.1 Vorteile der Key Indexed File Organisation
D.2 Prinzipielle Vorgehensweise zum Umkopieren von KIF's

0 EINLEITUNG

0.1 Grundlegende Charakteristika des System Command Interpreters

Der System Command Interpreter (SCI) ist die Schnittstelle zwischen dem Betriebssystem TAXO und dem Benutzer. Neben den bereits in anderen Handbüchern beschriebenen Möglichkeiten der bildschirmorientierten Bedienerführung sowie der Vorder- und Hintergrundverarbeitung besitzt der SCI einige bemerkenswerte Eigenschaften, die im folgenden kurz beschrieben werden sollen.

0.1.1 Definition der Terminals, an denen mit dem SCI gearbeitet werden kann

Normalerweise kann der SCI an jedem Terminal der TA 1600/30 aktiviert werden. Es besteht aber auch die Möglichkeit, ein Terminal so zu implementieren, daß es als reines Ein/Ausgabegerät verwendet wird, das keine Möglichkeit besitzt, mit TAXO zu kommunizieren.

Der SCI wird an einem Terminal durch das Kommando Q (QUIT) beendet. Nach Ausführung dieses Kommandos steht das Terminal als normales Ein/Ausgabegerät zur Verfügung.

Ein Terminal steht einem Anwenderprogramm außerdem zur Verfügung, wenn es mit dem Kommando XTS (Execute Task and Suspend SCI) gestartet wird. In diesem Fall sorgt TAXO dafür, daß der SCI solange mit der nächsten Eingabeaufforderung wartet, bis das Anwendungsprogramm beendet wird.

0.1.2 Login-Prozedur

Ein Benutzer muß beim Aufruf des SCI an einem Terminal seine Benutzerkennung mit Passcode eingeben, bevor er mit TAXO arbeiten kann. Wird diese Kennung als gültig angenommen, wird die Login-Prozedur aufgerufen. Diese Prozedur ist als SCI-Kommando mit dem Namen M$00 realisiert. In der TAXO-Standardversion wird geprüft, ob das System initialisiert ist. Sollte dies nicht der Fall sein, wird automatisch das IS (Initialize System) -Kommando aufgerufen und ausgeführt. Zum Schluß der Login-Prozedur wird die Anmeldung des Benutzers im Systemlogbuch protokolliert.

0.1.3 Logoff-Prozedur

Zum Beenden des SCI muß der Benutzer das Q (QUIT) -Kommando eingeben. Dieses Kommando ruft automatisch die Logoff-Prozedur. Diese Prozedur ist als SCI-Kommando mit dem Namen M$01 realisiert. Von M$01 wird momentan nur die Abmeldung des Benutzers im System-Logbuch protokolliert.

0.1.4 Privilegierung von Kommandos

Jedes SCI-Kommando erfordert zur Ausführung eine Mindestprivilegierungsstufe des Benutzers. Die Privilegierungsstufe (Privilege Level) eines Benutzers ist an seine Benutzerkennung gebunden und wird beim Einrichten der Kennung festgelegt. Ein Benutzer darf nur solche SCI-Kommandos ausführen, deren Privilege Level kleiner oder gleich dem Privilege Level seiner Benutzerkennung sind.

0.1.5 Synonyme

TAXO bietet dem Benutzer die Möglichkeit, für eine Zeichenkette eine Ersatz-Zeichenkette zu definieren. Diese Ersatz-Zeichenkette wird als Synonym bezeichnet und ist im allgemeinen kürzer als die Zeichenkette, die sie repräsentiert. Durch die Verwendung von Synonymen ist eine wesentliche Beschleunigung der Eingabe möglich. Alle von einem Benutzer definierten Synonyme sind an seine Benutzerkennung gebunden. Sie werden beim Logoff gespeichert und stehen beim nächsten Login wieder zur Verfügung.

Allen Programmen, die im Vordergrund laufen, stehen die Synonyme zur Verfügung. Beim Start eines Hintergrundprogramms wird diesem eine Kopie der aktuellen Synonyme zur Verfügung gestellt.

ANMERKUNG

Wird ein Synonym bei der Eingabe eines Pfadnamens verwendet, so kann es den ganzen Pfadnamen oder den ersten Teil des Pfadnamens repräsentieren. Ein Beispiel soll dies verdeutlichen:

Dem Synonym A ist der Pfadname des Directories VOL01.DIR1, dem Synonym B der Pfadname des Sub-Directories VOL1.DIR1.SUBDIR1 und dem Synonym C der Pfadname der Datei VOL1.DIR1.SUBDIR1.FILE1 zugewiesen. Folgende Eingaben bezeichnen die gleiche Datei:

A.SUBDIR1.FILE1
B.FILE1
C

0.1.6 Verzeichnis der TAXO SCI-Kommandos

In diesem Handbuch sind ausschließlich die Systemkommandos des Betriebssystems beschrieben. Alle Kommandos besitzen eine einheitliche Darstellung der Beschreibung:

- Format des Kommandos
- Benutzerantworten auf die Eingabeaufforderungen
- Ein Beispiel mit typischen Benutzerantworten und die entsprechende Ausführung des Kommandos.

Die Tabelle 0-1 enthält ein alphabetisches Verzeichnis aller verfügbaren SCI Kommandos und Angaben über die Kapitel, wo sie beschrieben sind.

0.2 Kommandoformat und Darstellungsform

Alle im Bedienerhandbuch angegebenen TAXO-Kommandos sind in einer einheitlichen Form beschrieben. In Abb. 0-1 ist diese Form am Beispiel des AF (Append File) -Kommandos gezeigt. Es werden nun die einzelnen Teile genauer beschrieben.

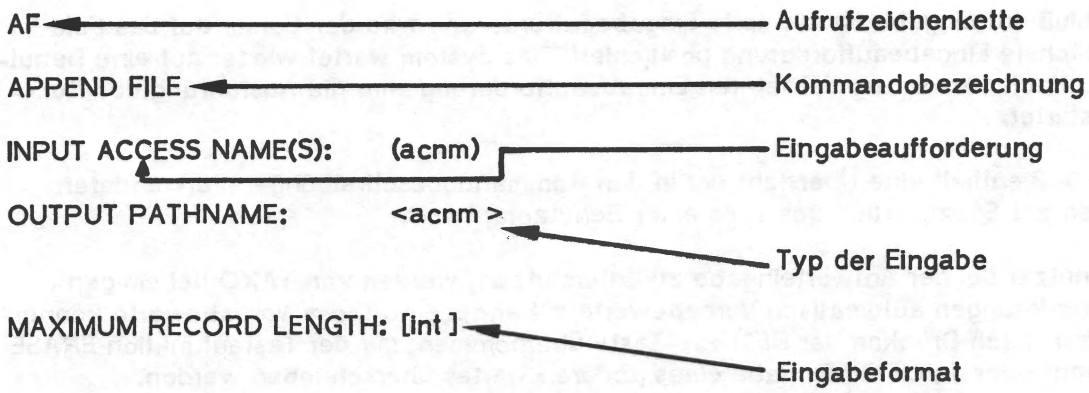


Abb. 0-1: Im Bedienerhandbuch 2 verwendete Form zur Kommandobeschreibung

0.2.1 Aufrufzeichenkette

Jedes Kommando wird durch Eingabe einer Aufrufzeichenfolge aufgerufen. Diese Zeichenfolge besteht meistens aus den Anfangsbuchstaben der englischen Kommandobezeichnung. Die Eingabe der Aufrufzeichenfolge ist mit der RETURN-Taste abzuschließen.

0.2.2 Kommandobezeichnung

Nach Eingabe der Aufrufzeichenkette antwortet TAXO mit der Kommandobezeichnung.

0.2.3 Eingabeaufforderung

Nach Eingabe eines Kommandos gibt TAXO neben der Kommandobezeichnung die zu dem Kommando gehörenden Eingabeaufforderungen am Bildschirm aus. Die Eingabeaufforderungen dienen dazu, den Benutzer über die zur Kommandoausführung notwendigen Parametereingaben zu informieren. Bei einigen Eingabeaufforderungen wird die Angabe eines Ein- bzw. Ausgabemediums verlangt. Dabei wird unterschieden zwischen Dateinamen, Gerätenamen und Zugriffsnamen. Auf die Eingabe eines Dateinamens wird in den Eingabeaufforderungen durch die englische Bezeichnung PATHNAME hingewiesen. Die Eingabe eines Gerätenamen wird durch die Bezeichnung DEVICE NAME verlangt. Ein Zugriffsname wird durch die Bezeichnung ACCESS NAME angefordert. In diesem Fall ist sowohl die Angabe eines Datei- als auch die eines Gerätenamen zulässig. Beispielsweise soll durch die Eingabeaufforderung OUTPUT PATHNAME angedeutet werden, daß für das Ausgabemedium ein Dateiname verlangt wird.

0.2.4 Typ der Eingabe

Nach Ausgabe der Eingabeaufforderungen am Bildschirm steht der Cursor hinter der ersten Eingabeaufforderung. Der Benutzer kann nun eine geeignete Antwort eintasten und mit der RETURN-Taste an das System senden.

Nach Abschluß der Eingabe für die erste Eingabeaufforderung wird der Cursor auf das Feld hinter die nächste Eingabeaufforderung positioniert. Das System wartet wieder auf eine Benutzereingabe. Nach Versorgung der letzten Eingabeaufforderung wird die Ausführung des Kommandos gestartet.

Die Tabelle 0-2 enthält eine Übersicht der in den Kommandobeschreibungen verwendeten Abkürzungen zur Spezifikation des Typs einer Benutzerantwort.

Um den Benutzer bei der Antworteingabe zu unterstützen, werden von TAXO bei einigen Eingabeaufforderungen automatisch Vorgabewerte mit angezeigt. Diese Vorgabewerte können vom Benutzer durch Drücken der RETURN-Taste übernommen, mit der Tastenfunktion ERASE FIELD gelöscht oder durch die Eingabe eines anderen Wertes überschrieben werden.

Bei den Vorgabewerten ist zwischen festen und variablen Werten zu unterscheiden. Bei festen Vorgabewerten wird bei jedem Kommandoaufruf immer der gleiche Wert angezeigt. Bei variablen Vorgabewerten werden eingegebene Werte von TAXO für einen späteren Aufruf des gleichen oder eines verwandten Kommandos gespeichert. Bis auf einige wenige Ausnahmen werden die variablen Vorgabewerte beim Abmelden vom System (QUIT) wieder gelöscht. Die nicht gelöschten sind an die Benutzerkennung gekoppelt und stehen nach dem nächsten Login wieder zur Verfügung.

Neben diesen Vorgabewerten gibt es bei vielen Kommandos Standardwerte für einzelne Parameter. Dies trifft insbesondere dann zu, wenn bei einer Eingabeaufforderung kein Wert eingetastet werden muß. Um den Standardwert bei einer Eingabeaufforderung auszuwählen, kann direkt die RETURN-Taste gedrückt werden, ohne daß vorher ein Zeichen eingegeben wurde. Eine solche Eingabe wird auch als leere Eingabe bezeichnet.

Tabelle 0-2: Antworttypen bei Eingabeaufforderungen und verwendete Abkürzungen

Abkürzung	Antworten
acnm	Der Name eines Gerätes (z.B. ST01; erlaubte Gerätenamen siehe Tabelle 0-3) oder einer Datei (z.B. VOL01.MYFILE)
int	Eine hexadezimale oder dezimale Ganzzahl oder ein ganzzahliger Ausdruck (z.B. 360/12). Hexadezimale Zahlen werden durch Eingabe einer führenden Null oder durch Voranstellen eines " >" Zeichens gekennzeichnet.
name	Eine alphanumerische Zeichenfolge beliebiger Länge. Das erste Zeichen muß ein Buchstabe sein.
string	Eine beliebige Folge von druckbaren Zeichen (z.B. "A=B"). Zeichenketten, die eines der folgenden Sonderzeichen enthalten, müssen in Ausführungszeichen (") eingeschlossen werden: = - Gleichheitszeichen @ - at sign & - kommerzielles Und * - Stern ! - Ausrufungszeichen , - Komma ' - Apostroph Ein Anführungszeichen innerhalb eines Strings muß durch zwei aufeinanderfolgende Anführungszeichen dargestellt werden.
yes/no	YES (für Ja) oder NO (für Nein). Außerdem erlaubt: beliebige Zeichenfolge beginnend mit Y - bedeutet Yes N - bedeutet No

Tabelle 0-3: Erlaubte Gerätenamen

Gerätename	Gerät
DSxx	Plattenlaufwerk Nummer xx xx im Bereich 01 bis 99 also: DS01, DS02, ..DS99
LPxx	Systemdrucker Nummer xx xx im Bereich 01 bis 99 also: LP01, LP02, ..LP99
ME	Das benutzereigene Terminal. Ein Benutzer kann mit Hilfe des STI (Show Terminal Information) -Kommandos den Namen seines eigenen Terminals bestimmen.
STxx	Terminal Nummer xx xx also im Bereich 01 bis 99 also: ST01, ST02, ..ST99 Ein Benutzer kann mit Hilfe des STI (Show Terminal Information) -Kommandos die Nummer seines eigenen Terminals bestimmen.
SPxx	Arbeitsplatzdrucker Nummer xx Ist dem Terminal STxx zugeordnet
MFDx	Mini Floppy Diskettenlaufwerk x x im Bereich 1-9. Zur Zeit wird nur ein Mini Floppy Diskettenlaufwerk in einem TAXO System unterstützt (MFD1).

0.2.5 Eingabeformat

In den Kommandobeschreibungen sind die zur Spezifikation des Typs der Antwort verwendeten Abkürzungen immer mit Sonderzeichen geklammert. Die Bedeutung dieser Symbole ist in Tabelle 0-4 beschrieben.

Es ist möglich, daß bei der Definition einer möglichen Antwort mehrere verschiedene Symbole verwendet werden. Ein Beispiel soll dies verdeutlichen:

> Durch die Angabe [(string)] wird ausgedrückt, daß die Eingabe wahlfrei ist. Eine getätigte Eingabe darf aus einer oder mehreren, durch Kommata getrennten, Zeichenketten (Strings) bestehen.

Tabelle 0-4: Symbole für Eingabeformatbeschreibung

Symbole	Bedeutung
< >	Die Antwort ist zwingend vorgeschrieben
[ ]	Die Antwort ist wahlfrei
{ }	Die Antwort muß einem der eingeklammerten Ausdrücke entsprechen. Sind die eingeklammerten Begriffe Zeichenketten in Großbuchstaben, muß eine dieser Zeichenketten angegeben werden.
( )	Die Antwort kann aus einer Liste vom vorgeschriebenen Typ bestehen. Die Angaben innerhalb der Liste sind durch Kommata getrennt.
@	In der Antwort ist die Verwendung von Synonymen zulässig.

1 IDENTIFIKATION UND SCHUTZ VON DATEIEN

1.1 Allgemeines

Dieses Kapitel beschreibt die Anwendung von Pfadnamen zur Identifikation von Dateien, den Gebrauch der Synonyme (Abkürzungen) für Pfadnamen, den Gebrauch alternativer Namen (engl. aliases) für Dateinamen, das Ändern von Dateinamen und wie man den Schutz von Dateien abändert.

1.2 Der Gebrauch von Pfadnamen (Pathnames)

Der Anwender ordnet Dateien, die er anlegt, Pfadnamen zu. Pfadnamen können sich aus folgenden Angaben, die eine Plattendatei kennzeichnen, zusammensetzen:

1. dem Namen der Platte, auf welchem die Datei liegt.
2. den Namen der Directories und Subdirectories, in welchen die Datei auf der Platte verzeichnet ist.
3. dem Namen der Datei.

Ein vollständiger Pfadname könnte zum Beispiel wie in Abb. 1-1 gezeigt aussehen.

Für die Länge der Pfadnamen gelten nur die folgenden Einschränkungen:

- Ein Pfadname besteht aus maximal 49 Zeichen.
- Befindet sich die Datei auf einer Sekundärplatte, so muß der Name dieser Platte eingefügt werden.

Die folgenden Unterabschnitte beschreiben, wie die einzelnen Teile eines Pfadnamens gebildet werden:

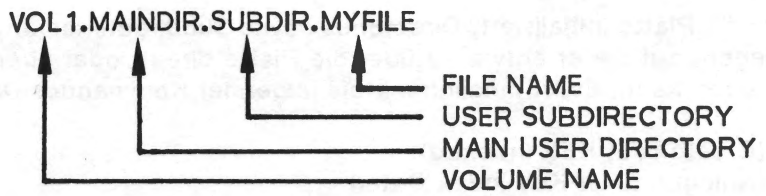


Abb. 1-1 Format eines vollständigen Pfadnamens.

1.2.1 Zusammensetzung von Plattennamen (VOLUME NAMES)

Das erste Element eines jeden Pfadnamens ist der Name der Platte. Plattennamen bestehen aus einer ein bis acht alphanumerischen Zeichen langen Zeichenkette, welche jene Platte kennzeichnet, auf welcher sich die Datei befindet. Das erste Zeichen des Plattennamens muß alphabetisch sein. Ein gültiger Pfadname könnte auch lediglich aus einem Plattennamen bestehen.

Platten erhalten ihren Namen durch das Kommando INV (Initialize New Volume) zugeteilt. Wurde eine Platte durch ein INV Kommando initialisiert, so muß bei jedem Zugriff auf eine Datei dieser Platte der Pfadname den Namen der Platte enthalten, es sei denn, die Platte ist die Systemplatte.

1.2.2 Bilden von Directories

Directory und Subdirectory sind Teil eines jeden Pfadnamens und werden durch das Kommando CFDIR (Create File Directory) angelegt. Kapitel 7 erklärt im Einzelnen, wie Directories und Subdirectories angelegt werden. Der Benutzer legt Directories und Subdirectories dann an, wenn er Dateien zusätzlich schützen oder in Gruppen zusammenfassen will.
Ein Subdirectory kann natürlich nur dann innerhalb eines Directories angelegt werden, wenn dieses Directory schon besteht.

So kann zum Beispiel das Kommando CFDIR ein Subdirectory mit dem Namen

VOL1.DIRA.DIRB

nur dann anlegen, wenn zuvor ein Directory mit dem Pfadnamen

VOL1.DIRA

angelegt worden war.

1.2.3 Anlegen von Dateien

Nachdem der Benutzer die Platte initialisiert, Directories sowie Subdirectories angelegt hat, kann er eine Datei anlegen, auf die er entweder über die Platte direkt, oder über ein Directory oder Subdirectory zugreifen kann. Es ist möglich mittels folgender Kommandos Dateien anzulegen:

- CFKEY (Create File, KEY indexed)
  (Anlegen einer Key Index Datei)
- CFREL (Create File, Relative record)
  (Anlegen einer Relative Record Datei)

- CFSEQ (Create File, SEQential)
  (Anlegen einer sequentiellen Datei)

- CFPRO (Create File, PROgram)
  (Anlegen einer Programm Datei)

- CFIMG (Create File, IMaGe)
  (Anlegen einer Image Datei)

- CF (Create File)
  (Anlegen einer Datei; erfordert nachfolgend die Angabe des Typs der Datei)

Damit werden die Pfadnamen festgelegt, durch welche die Dateien identifiziert werden. Kapitel 7 erklärt detailliert, wie Dateien angelegt werden.

Beispiel: Eine Datei, welche unter einem einstufigen Directory angelegt wurde, könnte durch einen Pfadnamen wie folgt bestimmt werden:

VOL1.JOESDIR.JOESFILE

1.3 Zuweisung von Synonymen (AS Kommando)

Der Benutzer kann eine Folge aus einem oder mehreren Zeichen als Ersatz für eine andere Zeichenkette definieren. Diese neue Zeichenkette kann (dann) bei Systemfragen anstelle der ursprünglichen Zeichenkette eingegeben werden.

Diese neudefinierte Zeichenkette nennt man ein Synonym, während die ursprüngliche Zeichenkette den Synonymwert angibt. So kann man zum Beispiel im Betriebssystem TAXO folgendes Synonym definieren:

Synonym	Synonymwert
DRUCKER	LP01
LOHNDATEI	VOL1.DIR2.FILE13

Gewöhnlich sind Synonyme kürzer als ihre Synonymwerte, gleichwohl können sie auch -wie oben gezeigt- länger sein. Sehr oft setzt man Synonyme für Pfadnamen ein. Ein Synonym kann sowohl für einen ganzen Pfadnamen stehen, als auch nur für den ersten Teil dieses Pfadnamens.

Beispiel: Steht der Buchstabe A als Synonym mit dem zugehörigen Synonymwert VOL1.DIR1.SUBDIR1 sowie MYFILE als Synonym mit dem Synonymwert VOL1.DIR1.SUBDIR1.FILE1, so gelten folgende Zeichenketten als gleichwertige Antworten auf Systemmeldungen:

MYFILE
A.FILE1
VOL1.DIR1.SUBDIR1.FILE1

Wurden Synonyme definiert, so verbindet TAXO das Synonym automatisch mit der Benutzerkennung. Um ein Synonym benutzen zu können, muß man unter jener Benutzerkennung arbeiten, unter welcher das Synonym zugewiesen wurde.

Die folgenden Kapitel beschreiben das Format des Kommandos AS (Assign Synonym), das die Zuweisung von Synonymen gestattet.

ANMERKUNG

Synonyme, die das System selbst benutzt, beginnen mit dem $ Zeichen. Daher sollten Benutzer ihre Synonyme nie mit dem $ Zeichen anfangen lassen.

1.3.1 Format des AS Kommandos

AS
ASSIGN SYNONYM VALUE
SYNONYM: <name>
VALUE: [string]

1.3.2 Benutzerantworten beim AS Kommando

Systemmeldung	Antwort erforderlich (R) o. optional (O)	Benutzerantworten
SYNONYM:	R	ein oder mehrere alphanumerische Zeichen (das erste muß immer alphabetisch und darf nicht das $ Zeichen sein). Bezeichnet den Namen des Synonyms, das zugewiesen werden soll.
VALUE:	O	die Zeichenkette, welche den Wert des Synonyms darstellen soll. Gibt der Benutzer keinen Wert für das Synonym ein, so wird der bisherige Wert des Synonyms gelöscht, das Synonym selbst getilgt.

1.3.3 Beispiel für das AS Kommando

Um dem Synonym VJ dem Pfadnamen VOL1.PROGDIR.DIRB.FILE1 das Synonym VJ zuzuweisen, muß der Benutzer das Kommando AS eingeben und auf die Meldungen des Systems wie folgt antworten:

```
( ) AS
    SYNONYM: VJ
    VALUE:   VOL1.PROGDIR.DIRB.FILE1
```

1.4 Anzeigen der Synonymwerte (LS Kommando)

Mit dem Kommando LS (List Synonyms) kann eine Liste aller Synonyme und Synonymwerte, die der Benutzerkennung zugeordnet sind, ausgedruckt, angezeigt, oder in einer Datei gespeichert werden.

1.4.1 Format des LS Kommandos

LS
LIST SYNONYMS
LISTING ACCESS NAME: [acnm] @

1.4.2 Benutzerantworten auf das Kommando LS

Systemmeldungen	Antwort erforderlich (R) o. optional (O)	Benutzerantwort
LISTING ACCESS NAME	O	der Name eines Gerätes, oder der Pfadname einer Datei, auf welche das System die Synonyme und deren Werte anzeigen, auflisten bzw. kopieren soll. Nur jene Synonyme und Synonymwerte können ausgegeben werden, die unter der betreffenden Benutzerkennung angelegt wurden.

1.4.3 Beispiel zu LS Kommando

Das folgende Beispiel zeigt, wie man mit dem LS Kommando eine Liste der Synonyme und Synonymwerte auf einem Terminal mit dem Gerätenamen ST09 angezeigt bekommen kann:

```
( ) LS
    LISTING ACCESS NAME: ST09
```

1.5 Modifizieren von Synonymen (MS Kommando)

Benutzer geben das Kommando MS (Modify Synonym) ein, um auf Synonyme und Synonymwerte, die mit ihrer Benutzerkennung verbunden sind, zugreifen zu können. TAXO antwortet auf die Eingabe eines MS Kommandos im VDT Modus durch Anzeigen des ersten verfügbaren Synonyms (in alphabetischer Reihenfolge) und dessen Wertes, wie in Abb. 1-2 gezeigt.

Im TTY Modus wird nur das Synonym ausgedruckt bzw. angezeigt.

SYNONYM: A

VALUE: VOL1.DIRA.FILE1

Abb. 1-2 Anzeige eines Synonyms und dessen Wert nach Eingabe eines MS-Kommandos

Erscheint das erste Synonym mit Synonymwert nach der Eingabe eines MS Kommandos, so kann der Benutzer ein neues Synonym und einen neuen Synonymwert eingeben oder er kann auch nur den Synonymwert verändern.

Synonyme können selbst nur durch Eingabe eines neuen Synonymwertes modifiziert werden.

Um ein Synonym und dessen Synonymwert zu modifizieren, sind die Tasten, wie in Tabelle 1-1 beschrieben, zu drücken. Tabelle 1-1 beschreibt die Tasten an Video Display Terminals (VDT).

Tabelle 1-1: Tasten, die man benutzt, um Synonyme und Synonymwerte am Video-Daten-Terminal (VDT) zu modifizieren.

Taste	Anwendung
CMD	MS Kommando wird verlassen.
Down ( ) Arrow	veranlaßt die Anzeige des (in alphabetischer Reihenfolge) nachfolgenden Synonyms.
RETURN	zeigt dem TAXO-Betriebssystem an, daß der Benutzer gerade ein Synonym oder einen Synonymwert modifiziert hat. Drückt man die RETURN Taste, ohne einen Synonymwert oder ein Synonym modifiziert zu haben, nimmt TAXO das (bisher) gültige Synonym oder den Synonymwert weiterhin an. Bei Bildschirmen im VDT Modus bedeutet im Betriebssystem TAXO das Beenden der Synonymmodifikation durch Drücken der RETURN Taste, daß der Cursor auf das Synonymwertfeld bewegt wird. Ist die Modifikation des Synonymwertes beendet, gibt TAXO das nächste Synonym mit Synonymwert aus. Sind keine weiteren Synonyme mehr definiert, werden die leeren Synonym- und Synonymwertfelder angezeigt. Im TTY Modus gibt das System nach der Synonymmodifikation den Synonymwert aus. Nach Ende einer Synonymwertmodifikation dagegen erscheint beim Betriebssystem TAXO das nächste Synonym, oder, falls es keine weiteren mehr gibt, ein leeres Synonymeingabefeld.
NEW LINE	Identisch mit RETURN Taste.
TAB/SKIP	gleiche Wirkung wie die RETURN Taste.
Up Arrow ( )	veranlaßt das System, das (in alphabetischer Reihenfolge) vorhergehende Synonym anzuzeigen.
SKIP	Synonym löschen.
ERASE FIELD	Das auf dem Display ausgegebene Feld löschen.

1.5.1 Format des MS Kommandos

MS
MODIFY SYNONYM

1.5.2 Benutzerantwort auf das MS Kommando

nicht erforderlich

1.5.3 Beispiel zum MS Kommando

Die nachfolgenden Bildschirmangaben zeigen an Hand des VDT das Abändern von Synonymen und Synonymwerten. Bediener von VDT's seien auf die Tabelle 1-1 verwiesen, um die Tasten an ihrem eigenen Bildschirm zu identifizieren, welche die in diesem Beispiel beschriebenen Funktionen ausführen. Um Platz zu sparen, werden die einzelnen Schritte, die der Benutzer macht, um ein Synonym bzw. einen Synonymwert zu ändern, nicht (explizit) angezeigt.

MS MODIFY SYNONYM	Der Benutzer gibt das MS Kommando ein.
SYNONYM: A VALUE: VOL1.DIR1.FILEA	TAXO zeigt das erstmögliche Synonym mit Synonymwert an.

Der Benutzer verändert das Synonym und drückt die RETURN Taste:

SYNONYM: AA VALUE:	TAXO zeigt das neue Synonym an und positioniert den Cursor an den Anfang des Synonymwertes, damit ein neuer Wert eingegeben werden kann.

Der Benutzer gibt den Synonymwert VOL1.DIRA.FILEAA ein und drückt die RETURN Taste.

SYNONYM: B VALUE: VOL1.DIRA.FILEB	Das nächste Synonym (in alphabetischer Reihenfolge) mit Synonymwert erscheint.

Der Benutzer drückt die RETURN Taste.

SYNONYM: B VALUE: VOL1.DIRA.FILEB	Der Cursor bewegt sich an den Anfang des Feldes Synonymwert, der nun verändert werden kann.

Der Benutzer gibt als Synonymwert VOL1.DIRB.FILEBB ein und drückt auf die RETURN Taste.

SYNONYM: C VALUE: VOL1.DIRA.FILEC	Das (alphabetisch) nächste Synonym und sein Synonymwert erscheinen am Bildschirm.

Der Benutzer drückt auf die Taste mit dem Pfeil aufwärts (↓)

SYNONYM: B VALUE: VOL1.DIRB.FILEBB	Das vorhergehende Synonym (nach dem Alphabet) und sein Wert werden angezeigt.

Der Benutzer drückt auf die Taste mit dem Pfeil aufwärts (↑)

SYNONYM: AA VALUE: VOL1.DIRA.FILEAA	Das (im Alphabet) vorhergehende Synonym und sein Synonymwert werden angezeigt.

Der Benutzer drückt die Taste "Pfeil abwärts" (↓)

SYNONYM: B VALUE: VOL1.DIRB.FILEBB	Das (in alphabetischer Reihenfolge) nächste Synonym und dessen Synonymwert werden angezeigt.

Der Benutzer drückt die COMMAND Taste um das MS Kommando zu verlassen.

Das MS Kommando gestattet weiterhin folgendes:

- Der Benutzer kann das nach dem Alphabet erste Synonym/Synonymwert Paar erhalten, indem er das Synonymfeld löscht und die Taste "Pfeil aufwärts" (↑) drückt.
- Der Benutzer kann das nach dem Alphabet letzte Synonym/Synonymwert Paar erhalten, indem er das Feld Synonym löscht und die Taste "Pfeil abwärts" (↓) drückt.
- Ist das Feld Synonym leer, kann der Benutzer nicht mit TAB über das Feld hinweggehen. In das Feld muß eingetastet werden, ansonsten ignoriert TAXO die TAB-Taste.
- Löscht der Benutzer das Feld Value und drückt die TAB-Taste, werden sowohl Synonym als auch Synonymwert gelöscht und das alphabetisch nachfolgende Synonym/Synonymwert Paar wird angezeigt.

1.6 Löschen aller Synonyme (DAS Kommando)

Mit Hilfe des DAS (Delete All Synonyms) Kommandos ist es möglich, alle zum Zeitpunkt des Kommandoaufrufs definierten und nicht vom System benötigten Synonyme zu löschen. Zu den vom System benötigten Synonymen zählen:

Synonymname	Verwendung
$$12	Kennzeichnung der verwendeten CPU
$$CC	Completion Code
$$DA	Kennzeichen für Debugger aktiviert
$$EA	Kennzeichen für Text Editor aktiviert
$$MO	Kennzeichen für Background/Foreground
$$ST	Terminalnummer
$$UI	Benutzerkennung (User ID)
ME	Terminalname

Der Benutzer hat beim Kommandoaufruf die Möglichkeit, eine Liste von Synonymen anzugeben, die durch DAS nicht gelöscht werden soll.

1.6.1 Format des DAS Kommandos

DAS E = (name, name,...)
DELETE ALL SYNONYMS

1.6.2 Benutzerantworten zum DAS Kommando

Das DAS Kommando verlangt keine Benutzerantworten. Beim Kommandoaufruf kann jedoch durch die Angabe

[EXCEPT] = (name, name,...)

eine Liste von Synonymen angegeben werden, die nicht gelöscht werden sollen.

1.6.3 Beispiel zum DAS Kommando

Nach der Fehlermeldung SYNONYM TABLE OVERFLOW will ein Benutzer alle Synonyme außer den Synonymen A und B löschen.

```
() LS
LIST SYNONYM
LISTING ACCESS NAME:

LIST SYNONYM
SYNONYM                    VALUE

 ¤¤12                       N
 ¤¤BC                       0C000
 ¤¤CC                       00000
 ¤¤DA                       N
 ¤¤EA                       N
 ¤¤MO                       OF
 ¤¤OB                       TAXOS130. DAS. LINKOBJ. DAS
 ¤¤PF                       . TA¤PROG
 ¤¤S                        TAXOS130. DAS. SOURCE. DAS
 ¤¤ST                       02
 ¤¤UI                       GGE022
 ¤¤VN                       TAXOS130
 ¤E¤C                       0
 ¤IT¤T                      DAS
 ¤XLE¤C                     TAXOS130. DAS. CONTRL. DAS
 ¤XLE¤L                     TAXOS130. DAS. MAP. DAS
 ¤XLE¤OB                    TAXOS130. DAS. LINKOBJ. DAS
 ¤XMA¤E                     TAXOS130. DAS. ERROR. DAS
 ¤XMA¤L                     TAXOS130. DAS. LIST. DAS
 ¤XMA¤O                     TAXOS130. DAS. OBJECT. DAS
 ¤XMA¤OP                    R, F
 A                          TAXOS130. DAS. DOCU. USERINST
 B                          TAXOS130. DAS. DOCUL. USERINST
 ME                         ST02
09:37:40 FRIDAY, JUL 09, 1982.

() DAS E=(A,B)
DELETE ALL SYNONYMS

() LS
LIST SYNONYM
LISTING ACCESS NAME:

LIST SYNONYM
SYNONYM                    VALUE

 ¤¤12                       N
 ¤¤CC                       00000
 ¤¤DA                       N
 ¤¤EA                       N
 ¤¤MO                       OF
 ¤¤ST                       02
 ¤¤UI                       GGE022
 A                          TAXOS130. DAS. DOCU. USERINST
 B                          TAXOS130. DAS. DOCUL. USERINST
 ME                         ST02
09:37:40 FRIDAY, JUL 09, 1982.

()
```

1.7 Zuordnung alternativer Namen (ALIASES) zu Namen von Directories oder Dateien (AA Kommando)

Das Kommando AA (Add Aliases) ordnet Dateien oder Directories alternative Namen zu. Als Antwort auf Systemmeldungen des AA Kommando gibt der Benutzer den Pfadnamen einer Datei oder eines Directories sowie einen zweiten Pfadnamen an, der als letztes Element den alternativen Namen enthält, durch welchen die Datei bzw. das Directory ebenfalls identifiziert werden kann. Beide Pfadnamen müssen identisch sein, bis auf die Dateinamen, welche jeweils das letzte Element der Pfadnamen darstellen.

1.7.1 Format des AA Kommandos

AA
ADD ALIAS TO PATHNAME
PATHNAME: <acnm> @
ALIAS PATHNAME: <acnm> @

1.7.2 Benutzerantworten beim AA Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	ein Pfadname für eine Datei oder ein Directory
ALIAS PATHNAME:	R	der gleiche Pfadname wie oben, nur für das letzte Element den alternativen Namen, durch welchen die Datei oder das Directory (ebenfalls) identifiziert werden können.

1.7.3 Beispiel zum AA Kommando

Um den alternativen Namen FILEB einer Datei mit Namen FILE1 zuzuordnen, muß der Benutzer das AA Kommando eingeben und wie folgt auf die Meldung des Systems antworten:

( ) AA
PATHNAME: VOL1.PROGDIR.DIRA.FILE1
ALIAS PATHNAME: VOL1.PROGDIR.DIRA.FILEB

1.8 Löschen alternativer Namen (DA Kommando)

Mit dem Kommando DA (Delete Aliases) kann die Zuweisung alternativer Namen aufgehoben werden. Ein als alternativer Name eingegebener Pfadname wird dadurch gelöscht.

1.8.1 Format des DA Kommandos

DA
DELETE ALIAS FROM PATHNAME
PATHNAME: <acnm > @

1.8.2 Benutzerantwort auf das DA Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	ein Pfadname, der den alternativen Namen enthält und gelöscht werden soll. Der alternative Name ist das letzte Element des Pfadnamens.

1.8.3 Beispiel zum DA Kommando

Um die Zuordnung des alternativen Namens FILEB zu einer Datei namens FILE1 (siehe Beispiel zum AA Kommando) rückgängig zu machen, gibt der Benutzer das DA Kommando (Delete Alias) ein und antwortet auf die Systemmeldung wie folgt:

```
( ) DA
    PATHNAME: VOL1.PROGDIR.DIRA.FILEB
```

1.9 Ändern von Dateinamen (MFN Kommando)

Das Kommando MFN (Modify File Name) ändert den Dateinamen und ordnet die mit der alten Datei verbundenen alternativen Namen dem neuen Dateinamen zu. Nach Eingabe des MFN Kommandos verlangt das System die Eingabe sowohl des bisher gültigen Pfadnamens der Datei als auch den neuen Pfadnamen, der von nun an die Datei identifizieren soll. Beide Pfadnamen sollten identisch sein, bis auf den Dateinamen, der als letztes Element in jedem Pfadnamen enthalten ist.

> ANMERKUNG
> Der Benutzer kann den Namen einer Datei, der eine LUNO zugeordnet ist, nicht verändern.

1.9.1 Format des Kommandos MFN

```
MFN
MODIFY FILE NAME
OLD PATHNAME:  < acnm >  @
NEW PATHNAME:  < acnm >  @
REPLACE?: <yes/no >
```

1.9.2 Benutzerantworten beim MFN Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
OLD PATHNAME:	R	der Pfadname, der die Datei augenblicklich identifiziert.
NEW PATHNAME:	R	der neue Pfadname, der nun die Datei identifizieren soll. Alle Elemente des Pfadnamens müssen identisch sein mit dem alten Pfadnamen, den neuen Dateinamen ausgenommen (das letzte Element im Pfadnamen).
REPLACE?:	R	Eingabe von Y (Yes) bewirkt die Ersetzung des alten Pfadnamens durch den neuen. Ist der neue Pfadname bereits einer vorhandenen Datei zugeordnet, wird diese Datei nach Ausführung des MFN Kommandos zerstört. Bei Eingabe N (No) wird der alte Pfadname nur dann durch den neuen Pfadnamen ersetzt, wenn der neue Pfadname nicht schon einer Datei zugeordnet ist. In diesem Fall (bei No) wird das Kommando MFN nicht ausgeführt, der Benutzer erhält eine Fehlermeldung.

1.9.3 Beispiel zum MFN Kommando

Das folgende Beispiel zeigt, wie man mit dem MFN Kommando die Datei VOL1.DIR2.FILE1 in VOL1.DIR2.FILEA umbenennen und den Inhalt der Datei VOL1.DIR2.FILEA dabei löschen kann.

```
( ) MFN
    OLD PATHNAME: VOL1.DIR2.FILE1
    NEW PATHNAME: VOL1.DIR2.FILEA
    REPLACE?: Y
```

1.10 Änderung des Schutzes einer Datei (MFP Kommando)

Das Kommando MFP (Modify File Protection) kann einer Datei Schreibschutz oder Löschschutz zuweisen, oder bereits existierenden Schutz einer Datei wieder aufheben. Alle Dateien werden ungeschützt angelegt und sie bleiben solange ungeschützt, bis ihnen ein MFP Kommando Schutz zuordnet.
Der Benutzer ordnet den jeweiligen Schutz durch Eingabe von Y (Yes) auf die entsprechenden Systemmeldungen zu, die auf das Kommando MFP hin ausgegeben werden. Um eine bereits vergebene Schutzart aufzuheben, gibt man N (No) auf die entsprechende Systemmeldung ein.

ANMERKUNG

Der Schutz einer Datei, welcher eine LUNO zugeordnet ist, kann vom Benutzer nicht verändert werden. Directories wie auch Datenfiles können mit diesem Kommando geschützt werden.

1.10.1 Format des MFP Kommandos

MFP
MODIFY FILE PROTECTION
PATHNAME: <acnm> @
WRITE-DELETE PROTECT?: <yes/no>
DELETE PROTECT?: <yes/no>

1.10.2 Benutzerantworten beim MFP Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Der Pfadname jener Datei, deren Schutz modifiziert werden soll.
WRITE-DELETE PROTECT?:	R	Eingabe von Y (YES) verleiht Schreib-/ Löschschutz. Eingabe von N (NO) hebt bereits vorhandenen Schutz einer Datei wieder auf.
DELETE PROTECT?:	R	Eingabe von Y (YES) sichert die Datei vor Löschung. Eingabe von N (NO) hebt bereits vorhandenen Löschschutz wieder auf.

1.10.3 Beispiel zum Kommando MFP

Um eine Datei, welche durch den Pfadnamen VOL1.DIRA.FILE2 identifiziert wird, vor Überschreiben und Löschen zu schützen, gibt der Benutzer das Kommando MFP ein und beantwortet die Systemmeldungen wie folgt:

```
( ) MFP
    PATHNAME: VOL1.DIRA.FILE2
    WRITE-DELETE PROTECT?: Y
    DELETE PROTECT?: N
```

1.11 Zuweisen von logischen Namen (ALN Kommando)

Das ALN Kommando (Assign Logical Name) verbindet einen logischen Namen mit einem EA-Gerät oder einer Datei wie folgt:

- Es spezifiziert einen logischen Namen und seine Parameter für eine permanente Datei (resource type = PERM)
- Es spezifiziert einen logischen Namen und seine LUNO für eine DC spezifische Datei (resource type = DC)
- Es spezifiziert einen logischen Namen ohne Parameter für ein EA-Gerät (resource type = NONE)

Anmerkung:

Das ALN Kommando definiert ein Synonym, dessen Name der eingegebene logische Name ist und dessen Wert aus den eingegebenen Benutzerantworten zusammengestellt wird. Diese logischen Namen können nur von COBOL Anwedungsprogrammen verarbeitet werden. Ihre Verwendung bei Systemkommandos kann zu nicht vorhersagbaren Fehlern führen.

1.11.1 Format des ALN Kommandos

```
( ) ALN
    ASSIGN LOGICAL NAME
    LOGICAL NAME: <name>
    RESOURCE TYPE:    PERM/DC/NONE
```

Abhängig vom resource type werden folgende weitere Eingabeparameter verlangt:

- resource type NONE:

  ```
  ASSIGN LOGICAL NAME
  ACCESS NAME: <acnm> @
  ```

- resource type PERM:

  ```
  PERMANENT FILE PARAMETERS
  FILE PATHNAME: <acnm> @
  ACCESS:            EA/EW/SH/RO
  ```

- resource type DC:

  ```
  ASSIGN DC PARAMETERS
  LUNO: <int>
  ```

1.11.2. Benutzerantworten zum ALN Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LOGICAL NAME	R	Ein ein bis acht Zeichen (alphanumerisch; erstes Zeichen Buchstabe) langer Name, der einem EA Gerät, oder einer Datei zugeordnet werden soll.
RESOURCE TYPE	R	Typ des logischen Namens: PERM: eine permanente Datei, die von einem anschließend gestartetem COBOL Programm mit bestimmten Zugriffsrechten verarbeitet werden soll. NONE: eine permanente Datei, die ohne besondere Parameter bearbeitet werden soll. DC: eine Verbindung für Datenfernübertragung
ACCESS NAME	R	Zugriffsname (für ein Gerät oder eine Datei), der dem logischen Namen zugeordnet werden soll.
FILE PATHNAME	R	Zugriffsname einer Datei, der dem logischen Namen zugeordnet werden soll.
ACCESS	R	Zugriffsrecht auf die Datei: EA - exklusive all EW - exklusive write SH - shared RO - read only
LUNO	R	Hexadezimale Angabe der LUNO, die mit dem logischen Namen verwendet wird.

1.11.3 Beispiel zum ALN Kommando

Die Datei VOL1.DIRA.FILEB soll von einem COBOL Programm unter dem Namen DATA mit exklusivem Schreibzugriff verarbeitet werden.

```
( ) ALN
   ASSIGN LOGICAL NAME
      LOGICAL NAME: DATA
      RESOURCE TYPE: PERM
      FILE PATHNAME: VOL1.DIRA.FILEB
      ACCESS: EW
```

1.12 Löschen von logischen Namen (RLN Kommando)

Das Kommando RLN (Release Logical Name) hebt eine mit dem ALN (Assign Logical Name) Kommando durchgeführte Zuweisung eines logischen Namens wieder auf.

1.12.1 Format des RLN Kommandos

```
RLN
RELEASE LOGICAL NAME
   LOGICAL NAME: <name>
```

1.12.2 Benutzerantworten zum RLN Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
LOGICAL NAME	R	Der logische Name, dessen Zuweisung aufgehoben werden soll.

1.12.3 Beispiel zum RLN Kommando

Die im Beispiel 1.11.3 durchgeführte Zuweisung des logischen Namens DATA soll wieder aufgehoben werden:

```
( ) RLN
   RELEASE LOGICAL NAME
      LOGICAL NAME: DATA
```

2 AUSGABE VON DATEIEN UND DIRECTORIES AUF BILDSCHIRM ODER DRUCKER

2.1 Allgemeines

Dieses Kapitel beschreibt die folgenden Kommandos, die es den Benutzern ermöglichen, die Inhalte von Dateien und Directories auf Bildschirm oder Drucker auszugeben:

Kommando	engl. Bedeutung	Funktion
° LD	List Directory	Directory auflisten
° MD	Map Disk	Platteninhalt auflisten
° MKF	Map Key Indexed File	Key Index Datei Parameter auflisten
° MPF	Map Program File	Programmdateiinhalt auflisten
° LLR	List Logical Record	Logischen Satz auflisten
° SF	Show File	Datei anzeigen

Das Ausdrucken von Dateien sowie die Überwachung des Druckvorgangs geschieht mit dem TAXO Spooler, der folgende Kommandos zur Verfügung stellt:

Kommando	engl. Bedeutung	Funktion
° PF	Print File	Datei ausdrucken
° SOS	Show Output Status	Druckstatus anzeigen
° HO	Hold Output	Druck anhalten
° RO	Resume Output	Druck fortsetzen
° KO	Kill Output	Druck abbrechen
° MSD	Modify Spooler Device	Formularwechsel durchführen
° BPPF	Build Partial Print File	Ausschnitt aus einer Druckdatei erstellen
° S$PF	Show Printed Files	Anzeige von gedruckten Dateien
° D$PF	Delete Printed Files	Löschen von gedruckten Dateien
° LPWITDTH	Line Printer Print Width	Einstellen der Druckbreite
° IPF	International Print File	Druckdatei für Drucker mit Landes-spezifischen Zeichenvorrat aufbereiten

2.2 Inhalte von Directories auflisten (LD Kommando)

Man gibt das LD Kommando ein, um die Namen aller Dateien und Subdirectories eines Directory aufzulisten. Die Dateinamen innerhalb der Subdirectories werden nicht mit ausgegeben. Das LD Kommando kann Directories mit 3000 oder mehr Einträgen nicht verarbeiten.
Dieses Kommando kann man sowohl im Vordergrund als auch im Hintergrund laufen lassen.

2.2.1 Format des LD Kommandos

```
LD
LIST DIRECTORY
PATHNAME: <acnm> @
LISTING ACCESS NAME: [ acnm ] @
FOREGROUND (Y/N): <yes/no>
```

2.2.2 Antworten des Benutzers beim LD Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME	R	Pfadname des Directories, von dem Datei- und Subdirectory-Namen aufgelistet werden sollen.
LISTING ACCESS NAME	O	Name des Gerätes oder Pfadname der Datei, wo TAXO die Liste ausgeben soll. Standardwert ist der Terminal Local File des Terminals.
FOREGROUND (Y/N) :	R	Die Eingabe von Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N(NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

2.2.3 Beispiel zum LD Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das LD Kommando eingeben muß, um eine Liste der Dateinamen und Namen der Subdirectories in einem Directory zu erhalten:

```
( ) LD
LIST DIRECTORY
PATHNAME: TAXO.JON.OBJEKT
LISTING ACCESS NAME: ME
FOREGROUND (Y/N): YES
( )
```

```
                                  (2)  (1)                    (3)      (7)          (8)
DIRECTORY LISTING OF: .PHIL
  MAX # OF ENTRIES: 17     # OF ENTRIES AVAILABLE: 11

(4)  DIRECTORY  ALIAS OF  ENTRIES    LAST UPDATE              CREATION
     *PROGRAMS     *        31    01/12/81  16:07:18    01/07/81  09:33:01
     MEDIA         *        27    01/10/81  13:05:13    01/07/81  09:36:01
(5)
                                                  (12)   (13)   (14)   (15)  (16)
(6)  FILE       ALIAS OF  RECORDS    LAST UPDATE        FMT    TYPE    BLK   PROTECT
     *KALIAS      KIF
(9)  KIF          *          9    01/14/81  09:23:29   BS     N KEY   YES   WRT DEL
     SEQ1         *         51    01/14/81  08:21:05   BS     N SEQ   YES
(10) SEQ2         *         28    01/14/81  09:09:27   NBS    C SEQ   YES   DEL
     SEQ3         *          3    01/14/81  08:23:45   BS     C SEQ   YES   WRT
(11) SEQ4         *         82    01/14/81  08:26:11   NBS    N SEQ   YES
     REL1         *        111    01/14/81  10:25:11   BS     N REL   YES
     PROG1        *         56    01/14/81  11:14:02   BS     C PRO   YES
     IMG11        *         22    01/14/81  12:01:01   BS     N IMG   YES
09:09:43 WEDNESDAY, JAN 14, 1981.
```

Ausgabe durch LD-Kommando

Feld	Beschreibung
1. DIRECTORY LISTING OF	Name der Directory von dem der Inhalt ausgelistet wird.
2. MAX # OF ENTRIES	Die max. Anzahl von Subdirectories oder Dateien die in diesem Directory angelegt werden können.
3. # OF ENTRIES AVAILABLE	Die Anzahl freier Plätze für Subdirectories und Dateien, die in diesem Directory verfügbar sind.
4. DIRECTORY	Name der Subdirectory die aufgelistet wird.
5. ALIAS OF	ALIAS NAME einer Directory auf das entweder unter Verwendung des Directory Namens oder des ALIAS Namens zugegriffen werden kann.
6. ENTRIES	Maximale Anzahl von Subdirectories und Dateien, die in einem Directory vorhanden sein können. Die Anzahl wird automatisch durch das Betriebssystem auf die nächste Primzahl erhöht.
7. LAST DATE	Datum und Uhrzeit an dem das Directory zuletzt aktualisiert wurde.
8. CREATION	Datum und Uhrzeit an dem das Directory angelegt wurde.
9. FILE	Name der jeweiligen Datei in dem Directory, das aufgelistet wird.
10. ALIAS OF	ALIAS Name einer Datei auf die entweder unter Verwendung des Dateinamens oder ALIAS Namens zugegriffen werden kann.
11. RECORDS	Anzahl der logischen Sätze in der spezifizierten Datei.
12. LAST UPDATE	Das neueste Datum und Uhrzeit wann die spezifizierte Datei zuletzt aktualisiert wurde.
13. FMT	Gibt das Format der gespeicherten Daten auf der Datei an. BS bedeutet Format mit Leerstellenunterdrückung (Blank Suppression) NBS bedeutet Format ohne Leerstellenunterdrückung (No Blank Suppression)

Feld	Beschreibung
14. TYPE	Der erste Buchstabe ist entweder N oder C. N gibt an, daß die Datei nicht zusammenhängend ist (das bedeutet, daß sie expandierbar ist); C git an, daß die Datei zusammenhängend ist (das bedeutet, daß die Datei nicht expandierbar ist). Die nächsten drei Buchstaben kennzeichnen den jeweiligen Dateityp.

Dateityp	Bedeutung
SEQ	sequentielle Datei
KEY	indexsequentielle Datei (Key Index)
REL	Relative Datei
IMG	Image Datei (relative Datei, spezieller Gebrauch)
PRO	Programmdatei

15. BLK — Gibt an, ob eine Datei geblockt ist.

YES	bedeutet geblockt
NO	bedeutet ungeblockt

16. PROTECT — Zeigt die Form des Dateischutzes an:

Code	Bedeutung
WRT	Geschützt gegen Überschreiben (schließt den Schutz gegen Löschen mit ein).
DEL	Geschützt gegen Löschen
WRT DEL	Geschützt gegen überschreiben und Löschen.

2.3 Platteninhalte auflisten - MD Kommando

Das MD Kommando (Map Disk) listet Informationen auf über die Inhalte einer Platte, eines Directories oder einer Datei. Es ist möglich, dieses Kommando sowohl im Vordergrund als auch im Hintergrund abzuarbeiten. MD kann Directories mit 3000 oder mehr Einträgen nicht verarbeiten. Der Umfang der ausgegebenen Informationen hängt von der gewünschten Form ab, die der Benutzer im Kommando angibt. Abb. 2-1 zeigt die Informationen, die ausgegeben werden, wenn die (ausführliche) Möglichkeit "Long Form" im MD Kommando gewählt wird. Tabelle 2-1 beschreibt die Informationen des MD Kommandos, wenn das Format "Long Form" gewählt wurde. Die Elementnummern in Tabelle 2-1 beziehen sich auf die numerierten Elemente in Abb. 2-1.

Die Benutzer sollten beachten, daß der Directoryname VCATALOG, der in Plattenverzeichnissen erscheint, sich auf ein Directory bezieht, das vom Betriebssystem TAXO gepflegt wird. VCATALOG ist das Directory mit der höchsten Stufe oder der Stufe Null in jeder Platte. Durch Drücken der COMMAND Taste kann das MD Kommando vorzeitig abgebrochen werden.

2.3.1 MD Kommando: Wahlmöglichkeit SHORT FORM

Wenn der Benutzer die Option "Short Form" angibt, begrenzt er die Informationsausgabe des MD Kommandos auf die Elemente 1 bis 10 aus der Abb. 2-2. Diese Elemente sind weiter unten in diesem Kapitel, in Abb. 2-1, beschrieben.

2.3.2 MD Kommando: Wahlmöglichkeit TOP LEVEL ONLY

Die Option "Top Level only" erzeugt Informationen über alle Dateien und Subdirectories in dem angegebenen Directory. Die Inhalte von Subdirectories werden nicht aufgelistet.

Ein Ausgabebeispiel ist in Abb. 2-3 gezeigt.

2.3.3 MD Kommando: Wahlmöglichkeit DIRECTORY NODES ONLY

Gibt der Benutzer die Option "Directory Nodes only" an, so begrenzt er die Ausgabe des MD Kommandos auf die folgenden Informationen über die Directories einer Platte:

- Die Anzahl von Subdirectories und Dateien, die in jedem Directory angelegt werden.
- Die Anzahl freier Plätze für Dateien und Subdirectories, die noch in dem Directory verfügbar sind.
- Der gesamte Speicherplatz, der von dem Directory mit seinen Subdirectories und Dateien belegt wird.

Abb. 2-4 zeigt ein Beispiel der Informationen, die durch ein MD Kommando mit der Wahl "Directory Nodes only" erzeugt werden.

2.3.4 Format des MD Kommandos

```
MD
MAP DISK
PATHNAME: <acnm> @
LISTING ACCESS NAME: [ acnm ] @
SHORT FORM?: <yes/no>
TOP LEVEL ONLY?: <yes/no>
DIRECTORY NODES ONLY?: <yes/no>
FOREGROUND (Y/N): <yes/no>
```

```
( ) MD
MAP DISC
  PATHNAME: VOL1
  LISTING ACCESS NAME:
  SHORT FORM?: NO
  TOP LEVEL ONLY?: NO
  DIRECTORY NODES ONLY: NO
      FOREGROUND (Y/N): YES
```

```
        TODAY IS 16:22:11 TUESDAY, MAY 31, 1977.

                                    NUMBER  CURRENT  TOTAL
                                     OF      EOM    ALLOC
LV NAME                   FILE     RECORDS  ADU     ADU      LAST UPDATE
-- ------------------     TYPE     -------  -----   -----    ---------------
 0 VCATALOG:                                   FILES=6  AVAILABLE=338
     DIR1                  D   12        12      12      5/31/77  16:15:56

  ALLOC: PRI=12    SEC=0        #SECS=0   LRECL=134  PRECL=134  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NNNYN   DATA-FMT=BINARY    CREATED  5/ 5/77  15:12:10

     DIR2                  D   24        24      24      5/ 5/77  15:13:32

  ALLOC: PRI=24    SEC=0        #SECS=0   LRECL=134  PRECL=134  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NNNYN   DATA-FMT=BINARY    CREATED  5/ 5/77  15:13:31

     VCATALOG              D   348       0       348     1/ 0/ 0   0: 0: 0

  ALLOC: PRI=348   SEC=0        #SECS=0   LRECL=134  PRECL=134  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NYNYN   DATA-FMT=BINARY    CREATED  1/ 0/ 0   0: 0: 0

     FILE1                 S   0         0       34      5/ 5/77  15:47:57

  ALLOC: PRI=34    SEC=1        #SECS=0   LRECL=80   PRECL=288  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NNYYN   DATA-FMT=BINARY    CREATED  5/ 5/77  15:47:56

     FILE2                 R   0         0       144     1/ 0/ 0   0: 0: 0

  ALLOC: PRI=144   SEC=286      #SECS=0   LRECL=80   PRECL=560  ADU/BLK=2
 WPT/DPT/BLK/PERM/FORCED/=NYYYY   DATA-FMT=BINARY    CREATED  5/ 5/77  15:59: 9

     FILE3                 I   0         0       1000    1/ 0/ 0   0: 0: 0

  ALLOC: PRI=1000  SEC=0        #SECS=0   LRECL=288  PRECL=288  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=YYNYN   DATA-FMT=BINARY    CREATED  5/ 5/77  16: 2: 6

 1  DIR1:                                     FILES=1  AVAILABLE=9
     FILE1                 K             3014/   '68     5/31/77  16:15:58

  ALLOC: PRI=168   SEC=68       #SECS=0   LRECL=80   PRECL=560  ADU/BLK=2
 WPT/DPT/BLK/PERM/FORCED/=NNYYY   DATA-FMT=B SUPPRS  CREATED  5/31/77  16:15:56

**VOL1.DIR1                                   TOTAL SIZE = 180 ADUS
 1  DIR2:                                     FILES=1  AVAILABLE=22
     FILE2                 P   74        74      1000    5/ 6/77  16:28:25

  ALLOC: PRI=1000  SEC=500      #SECS=0   LRECL=288  PRECL=288  BLK/ADU=1
 WPT/DPT/BLK/PERM/FORCED/=NNNYN   DATA-FMT=BINARY    CREATED  5/ 6/77  16:28:22

**VOL1.DIR2                                   TOTAL SIZE = 1024 ADUS-
**VOL1                                        TOTAL SIZE = 2730 ADUS
```

Abb. 2-1: Ausgabe durch MD Kommando im Format "Long Form"

Tabelle 2-1:

Ausgabe durch MD Kommando im Format "Long form"

Feld	Beschreibung
1. LV	Jedes Directory und Subdirectory auf der Platte ist identifizierbar durch eine Stufennummer (Level), die die Stufe festlegt, auf der jedes von ihnen angelegt wurde.
2. NAME	Namen der Directories oder der Dateien, die sich auf der Platte oder in dem Directory von dem der Inhalt aufgelistet werden soll, befinden.
3. FILE TYPE	Die Dateitypen von Directories und Subdirectories werden folgendermaßen identifiziert:

Dateityp	Bedeutung
D	Directory (relative Datei, spezieller Gebrauch)
I	Imagedatei (relative Datei, spezieller Gebrauch)
K	Key Index Datei
P	Programmdatei
R	Relative Datei
S	Sequentielle Datei

4. NUMBER OF RECORDS	Die aktuelle Anzahl der logischen Sätze in der Datei. Bei relativen Dateien ist diese Anzahl die Nummer des Satzes mit der höchsten relativen Satznummer in der Datei. Dieses Feld bleibt bei Key Index Dateien die unter TAXO V 1.3.0 nach dem "Hashed Placement" Verfahren erstellt wurden, frei.
5. CURRENT EOM ADU	Die Anzahl von ADUs (Allocatable Disk Units), die benötigt wird, um den aktuellen Datenbestand aufzunehmen.
6. TOTAL ALLOC ADU	Die Gesamtzahl von ADUs, die für das Directory oder die Datei angelegt ist.
7. LAST UPDATE	Datum, an dem das Directory oder die Datei zuletzt aktualisiert wurde (Monat/Tag/Jahr, Stunde:Minute:Sekunde).
8. FILES =	Die Anzahl von Subdirectories und Dateien, die in dem Directory angelegt worden sind.
9. AVAILABLE =	Die Anzahl freier Plätze für Subdirectories und Dateien, die in dem Directory noch verfügbar sind.
10. TOTAL SIZE =	Gesamter Speicherplatz (in ADUs), der aktuell für alle Dateien in dem Directory, das links angegeben ist, angelegt ist.
11. PRI =	Größe (in ADUs) eines File oder Directory nach dem ersten Anlegen.

Feld	Beschreibung
12. SEC =	Größe (in ADUs) eines File oder Directory für Erweiterungen. Null wird angegeben, wenn die Datei nicht expandierbar ist.
13. #SECS =	Anzahl von Erweiterungen, die für eine Datei durchgeführt wurden.
14. LRECL	Länge des logischen Satzes (in Bytes), die der Datei zum Zeitpunkt des Anlegens zugewiesen wurde.
15. PRECL	Länge des physikalischen Satzes (in Bytes), die der Datei zum Zeitpunkt des Anlegens zugewiesen wurde.
16. ADU/BLK (oder BLK/ADU)	Beziehungen zwischen physischen Sätzen (Blöcken) und ADU-Größe. ADU/BLK ist die Anzahl von ADUs in einem Block, oder BLK/ADU ist die Anzahl von Blöcken in einer ADU.
17. WPT/DPT/ BLK/PERM/ FORCED	zeigt, ob die folgenden Dateizustände gegeben sind: - WPT - Write Protected - geschützt gegen Schreiben - DPT - Delete Protected - geschützt gegen Löschen - BLK - geblockt - PERM - permanent - FORCED - Forced Write - erzwungenes Schreiben Ein Y (YES) (in einer Kette von Y's und N's) erscheint, wenn die spezielle Bedingung erfullt ist, ein N, wenn sie nicht gegeben ist.
18. CREATED	Datum, an dem die Datei angelegt wurde. (Monat/Tag/Jahr, Stunde:Minute:Sekunde).
19. DATA-FMT	Format der Daten, die in der Datei gespeichert sind. B SUPPR zeigt an, daß die Daten im Format Leerzeichen-Unterdrückung aufgezeichnet sind, BINARY zeigt an, daß die Daten in der Datei im Format ohne Leerzeichenunterdrückung gespeichert sind.

```
( ) MD
MAP DISC
  PATHNAME: VOL1
  LISTING ACCESS NAME:
  SHORT FORM?: YES
  TOP LEVEL ONLY?: NO
  DIRECTORY NODES ONLY?: NO
  FOREGROUND (Y/N): YES

      TODAY IS 16:19:05 TUESDAY, MAY 31, 1977.

                                        NUMBER  CURRENT  TOTAL
                                  FILE    OF      EOM    ALLOC
  LV NAME                         TYPE  RECORDS   ADU     ADU       LAST UPDATE
  -- ------------------            -    ------   -----   -----    ---------------
   0 VCATALOG:                                     FILES=6  AVAILABLE=338
       DIR1                        D     12       12      12       5/31/77  16:15:56
       DIR2                        D     24       24      24       5/ 5/77  15:13:32
       VCATALOG                    D     348      0       348      1/ 0/ 0   0: 0: 0
       FILE1                       S     0        0       34       5/ 5/77  15:47:57
       FILE2                       R     0        0       144      1/ 0/ 0   0: 0: 0
       FILE3                       I     0        0       1000     1/ 0/ 0   0: 0: 0
   1  DIR1:                                      FILES=1  AVAILABLE=9
        FILE1                      R              3014702 168      5/31/77  16:15:58
  **VOL1.DIR1                                      TOTAL SIZE = 180 ADUS
   1  DIR2:                                       FILES=1  AVAILABLE=22
        FILE2                      P     74       74      1000     5/ 6/77  16:28:25
  **VOL1.DIR2                                      TOTAL SIZE = 1024 ADUS
  **VOL1                                           TOTAL SIZE = 2730 ADUS

  []
```

Abb. 2-2: Ausgabe durch MD Kommando im Format "Short Form".

```
( ) MD
MAP DISC
  PATHNAME: VOL1
  LISTING ACCESS NAME:
  SHORT FORM?: YES
  TOP LEVEL ONLY?: YES
  DIRECTORY NODES ONLY?: NO
  FOREGROUND (Y/N): YES

      TODAY IS 20:18:38 WEDNESDAY, JUN 29, 1977.

                                        NUMBER  CURRENT  TOTAL
                                  FILE    OF      EOM    ALLOC
  LV NAME                         TYPE  RECORDS   ADU     ADU       LAST UPDATE
  -- ------------------            -    ------   -----   -----    ---------------
   0 VCATALOG:                                     FILES=6  AVAILABLE=338
       DIR1                        D     12       12      12       5/31/77  16:15:56
       DIR2                        D     24       24      24       5/ 5/77  15:13:32
       VCATALOG                    D     348      0       348      1/ 0/ 0   0: 0: 0
       FILE1                       S     0        0       34       5/ 5/77  15:47:57
       FILE2                       R     0        0       144      1/ 0/ 0   0: 0: 0
       FILE3                       I     0        0       1000     1/ 0/ 0   0: 0: 0
  **VOL1                                           TOTAL SIZE = 2730 ADUS

  []
```

Abb. 2-3: Ausgabe durch MD Kommando im Format "Top Level only".

```
( ) MD
MAP DISC
  PATHNAME: VOL1
  LISTING ACCESS NAME:
  SHORT FORM?: YES
  TOP LEVEL ONLY?: NO NO
  DIRECTORY NODES ONLY?: YES
  FOREGROUND (Y/N): YES
        TODAY IS 16:49:31 TUESDAY, MAY 31, 1977.

  0 VCATALOG:                                   FILES=6  AVAILABLE=338
  1   DIR1:                                     FILES=1  AVAILABLE=9
**VOL1.DIR1                                     TOTAL SIZE = 180 ADUS
  1   DIR2:                                     FILES=1  AVAILABLE=22
**VOL1.DIR2                                     TOTAL SIZE = 1024 ADUS
**VOL1                                          TOTAL SIZE = 2730 ADUS


[]
```

Abb. 2-4: Ausgabe durch MD Kommando im Format "Directory Nodes only".

2.3.5 Antworten des Benutzers beim MD Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Pfadname, der die Platte, das Directory höchster Stufe oder die Datei identifiziert, über die der Benutzer Informationen haben will.
LISTING ACCESS NAME:	O	Gerätename oder Pfadname einer Datei, in der die Informationen aufgelistet werden sollen, die das MD Kommando erzeugt. Standardwert ist der Terminal Local File des Terminals. Wird das Kommando im Hintergrund abgesetzt, so ist hier die Eingabe von 'ME' nicht gestattet.
SHORT FORM:	R	Y (YES) zeigt an, daß die Information nur im Format "Short Form" (wie in Tab. 2-1 beschrieben) ausgegeben werden soll, N (NO) bezeichnet die "Long Form" (s. Tab. 2-1).
TOP LEVEL ONLY?:	R	Y (YES) zeigt an, daß nur Informationen über das Directory aufgelistet werden sollen, die in der Antwort auf die System-Ausgabe "PATHNAME:" aufgeführt wurden. N (NO) zeigt an, daß die Liste auch Informationen über alle Files und Subdirectories in dem Directory beinhalten soll.

Systemausgaben	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
DIRECTORY NODES ONLY?:	R	Y (YES) bewirkt die Ausgabe von Informationen nur über Directories (nicht über Dateien) in der folgenden Weise: 1. Die Anzahl der Subdirectories und Dateien, die in dem Directory angelegt sind. 2. Die Anzahl freier Plätze, die noch zum Anlegen von Dateien und Subdirectories in diesem Directory verfügbar sind. N (NO) besagt, daß Informationen über alle Dateien in dem Directory bereitgestellt werden sollen.
FOREGROUND (Y/N)	R	Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. N(NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

2.3.6 Beispiel zum MD Kommando

Die Abbildungen 2-1 bis 2-4 weiter oben in diesem Kapitel zeigen Beispiele der Eingabe des MD Kommandos und die Listen, die durch die Eingabe verschiedener Wahlklauseln erstellt werden können.

2.4 Auflistung der Schlüsselbeschreibungen von Key Index Dateien - MKF Kommando

Man ruft das MKF Kommando (Map Key Index Datei), um die Charakteristika einer Key Index Datei angezeigt zu bekommen.

Abb. 2-5 zeigt das Format der angezeigten Informationen.
Tab. 2-2 beschreibt die Information, die von MKF ins Listing ausgegeben wird.

2.4.1 Format des MKF Kommandos

MKF
MAP KEY INDEX FILE
PATHNAME: <acnm> @
LISTING ACCESS NAME: [acnm] @

2.4.2 Benutzerantworten beim MKF Kommando

Systemausgaben	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Pfadname, der die Key Index Datei identifiziert, die aufgelistet werden soll.
LISTING ACCESS NAME:	O	Gerätename oder Pfadname einer Datei, wohin TAXO die Liste für die angesprochene Datei ausgeben soll. Standardwert ist der Terminal Local File des Terminals.

```
() MKF
MAP KEY INDEX FILE
 PATHNAME:    VOL1.DIR1.FILE1
 LISTING ACCESS NAME:    ME
     FILE MAP OF VOL1.DIR1.FILE1
       TODAY IS 16:17:34 TUESDAY, MAY 31, 1977.

KEYS:
          START                                DUPLICATES
  KEY     COLUMN     LENGTH     MODIFIABLE      ALLOWED
    1        1          5          N              N
    2        8          3          Y              Y
    3       25         15          Y              N


()
```

Abb. 2-5: Durch MKF Kommando angezeigte Information über eine Key Index Datei.

2.4.3 Beispiel zum MKF Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das MKF Kommando aufrufen kann, um eine Liste einer Key Index Datei zu erhalten.

```
( ) MKF
MAP KEY INDEXED FILE
PATHNAME: VOL1.DIR1.FILE1
LISTING ACCESS NAME: ME
FILE MAP OF   VOL1.DIR1.FILE1
( )
```

Tabelle 2-2: Beschreibung der Ausgabe durch MKF Kommando.

Überschrift	Beschreibung
KEY	Eine Zahl zwischen 1 und 14, die jeden Schlüssel der Datei bezeichnet. Schlüssel Nr. 1 ist der primary Key (Primarschlüssel).
START COLUMN	Zeichenposition in einem Satz, an der ein Schlüsselfeld beginnt. Die erste Zeichenposition im Satz ist 1.
LENGTH	Länge eines Schlüsselfeldes.
MODIFIABLE ?	N (NO) besagt, daß der Inhalt eines Schlüsselfeldes vorhanden sein muß und nicht verändert werden darf, wenn der Satz, der diesen Schlüssel enthält, zurückgeschrieben wird. Y (YES) zeigt an, daß der Inhalt eines Schlüsselfeldes sich ändern kann, wenn der dazugehörige Satz zurückgeschrieben wird, und daß er nicht unbedingt vorhanden sein muß, wenn der Satz geschrieben wird.
DUPLICATES ALLOWED	Y (YES) zeigt an, daß der Wert des Schlüssels im gleichen Schlüsselfeld bei mehreren Sätzen in der Datei doppelt vorhanden sein kann. N (NO) besagt, daß der Wert dieses Schlüssels im gleichen Feld verschiedener Sätze nur einmal vorhanden sein darf.

2.5 Auflisten des Inhalts einer Programmdatei - MPF Kommando

Man gibt das MPF Kommando ein (Programmdatei auflisten), um Informationen über Tasks, Prozeduren und Overlays zu erhalten, die in einer Programmdatei enthalten sind.

Abb. 2-6 zeigt das Format der ausgegebenen Informationen.
Tab. 2-3 beschreibt die Informationen, die von MPF für die Programmdatei ausgegeben werden.

2.5.1 Format des MPF Kommandos

```
MPF
MAP PROGRAM FILE
PATHNAME: <acnm> @
LISTING ACCESS NAME: [ acnm ] @
```

2.5.2 Benutzerantworten beim MPF Kommando

Systemausgaben	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Pfadname, der die Programmdatei identifiziert, die aufgelistet werden soll.
LISTING ACCESS NAME:	O	Name eines Gerätes oder Pfadname einer Datei, wohin das Betriebssystem TAXO die Liste der angegeenen Datei ausgeben soll. Standardwert ist der Terminal Local File des Terminals.

```
      FILE MAP OF .S$PROGA
        TODAY IS 00:00:41
TASKS:     MAXIMUM POSSIBLE = 255
 ID    NAME     LENGTH  LOAD  PRI   S P M R D E O C  OVLY  P1/SAME P2/SAME  INSTALLED
 01  RTT001      001E   0000    4         R                                   5/31/78
 02  SCILMT      0136   C000    0   S P   R D              01/N              4/24/78
 04  OQ$COPY     0414   0600    2         R D              02/Y     04/Y     4/24/78
PROCEDURES:    MAXIMUM POSSIBLE = 255
 ID    NAME     LENGTH  LOAD      M D E W C                                 INSTALLED
 02  SCA         0580   0000      M D                                          4/24/78
 05  TXDXFC      1D94   0000        D                                          4/24/78
OVERLAYS:   MAXIMUM POSSIBLE = 255
 ID    NAME     LENGTH  LOAD  MAP   D   OVLY                                 INSTALLED
 02  PARSER      1760   4A56        D                                          4/24/78
 03  BUILD       1F66   1CCE        D    05                                    4/24/78
```

Abb. 2-6 Anzeige der Information durch MPF Kommando.

Tabelle 2-3 Programm File Informationsausgabe durch ein MPF-Kommando

Überschrift	Beschreibung
ID	Die Identifikationsnummer, die einer Task, einer Prozedur oder einem Overlay zugewiesen wurde, als diese installiert wurde.
NAME	Der Name einer Task, Prozedur oder eines Overlays.
LENGTH	Die Größe einer Task, Prozedur oder eines Overlays in Bytes, die im Speicher beansprucht werden.
LOAD	Die natürliche Ladeadresse einer Task, Prozedur oder eines Overlays (der Anfang wird in der Link Editor Liste angezeigt).
PRI	Die Priorität einer Task
SPMRDEOC (für Tasks)	Die entsprechenden Zeichen für die Eigenschaften der Task sind nachfolgend beschrieben:

Spalte Bezeichnung	Eigenschaft der Task
S	Die Task ist eine Systemtask
P	Die Task ist privilegiert
M	Die Task ist speicherresident
R	Die Task ist replicatibel
D	Die Task ist vor Löschung geschützt
E	Die Task kann nicht ausgeführt werden
O	Die Task ist vor einem Overflow geschützt und wird im Falle eines arithmetischen Overflows mit Task Error Code >F beendet.
C	Die Task benutzt die vordefinierten XOP's im Schreibspeicher.

OVLY (nur für Tasks)	Die ID-Nummer eines Overlays, das zuletzt mit einer Task verbunden war.
P1/SAME	Eine Nummer in der P1-Spalte stellt die installierte ID-Nummer der ersten Prozedur dar, die mit einer Task verbunden ist. Die Y (yes)-Angabe in der SAME-Spalte zeigt an, daß die in der P1-Spalte identifizierte Prozedur vom gleichen Programm-File stammt. Die Angabe N (no) zeigt an, daß die Prozedur zum augenblicklichen System-Programm-File gehört.
P2/SAME	Eine Nummer in der P2-Spalte ist die installierte ID-Nummer der zweiten Prozedur, die mit einer Task verbunden war. Die Y (yes)-Angabe in der SAME-Spalte zeigt an, daß die in der P2-Spalte identifizierte Prozedur vom gleichen Programm-File ist. Die Angabe N (no) zeigt an, daß die Prozedur zum augenblicklichen System-Programm-File gehört.

Tabelle 2-3 Programm File Informationsausgabe durch ein MPF-Kommando

Überschrift	Beschreibung
INSTALL DATE	Das Datum, an dem die Task, Prozedur oder das Overlay eingerichtet wurde. Das Format, in dem das Datum und die Zeit angezeigt werden, ist folgendes: Monat/Tag/Jahr Stunde/Minute/Sekunde z.B. 03/ 12/ 77/ 00/ 00/ 00 Die Zeit ist immer Null, da TAXO keine Zeit in Programm-Files speichert.
RES	Zeigt an, daß eine Prozedur speicherresident ist.
D	Zeigt an, daß die Prozedur vor Löschung geschützt ist.
E	Zeigt an, daß die Prozedur vor Ausführung geschützt ist.
W	Zeigt an, daß die Prozedur schreibgeschützt ist.
C	Zeigt an, daß die Prozedur die vordefinierten XOP's im Schreibspeicher benutzt.
MAP	Zeigt an, daß das Overlay mit einer Verschiebungsbitliste verbunden ist.
D	Zeigt an, das das Overlay vor Löschung geschützt ist.
OVLY (für Overlays)	Wenn ein anderes Overlay mit einer Task verbunden ist, erscheint die ID-Nummer dieses Overlays in dieser Spalte.

2.5.3 Beispiel zum MPF Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein MPF-Kommando eingeben kann, um einen Ausdruck eines Programm-Files zu erhalten.

```
( ) MPF
MAP PROGRAM FILE
   PATHNAME: J.PROG
   LISTING ACCESS NAME: LP01
   ( )
```

2.6 Auflisten der Inhalte logischer Sätze (LLR Kommando)

Der Anwender gibt das LLR-Kommando (List Logical Record) ein, um Inhalte eines Satzes oder mehrere Sätze, die in einer Datei stehen, auflisten zu lassen. Die Inhalte der angeführten Sätze werden in zwei Formaten ausgedrückt: hexadezimaler und ASCII-Code-Ausdruck.

Es besteht die Möglichkeit, dieses Kommando sowohl im Vordergrund als auch im Hintergrund abzuarbeiten.

2.6.1 Format des LLR Kommandos

```
LLR
LIST LOGICAL RECORD
PATHNAME: <acnm> @
STARTING RECORD: <int>
NUMBER OF RECORDS: [ int ]
LISTING ACCESS NAME: [ acnm ] @
FOREGROUND (Y/N): <yes/no>
```

2.6.2 Antworten des Benutzers beim LLR Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Antwort des Anwenders
PATHNAME:	R	Ein Pfadname, der die Datei bezeichnet, deren Sätze ausgedruckt werden sollen.
STARTING RECORD:	O	Eine Dezimal- oder Hexadezimalganzzahl, die den ersten Satz bezeichnet, dessen Inhalt gelistet werden soll. Eine fehlende Antwort gilt als 0.
NUMBER OF RECORDS:	O	Eine Dezimal- oder Hexadezimalzahl, die angibt, wie viele Sätze gelistet werden sollen. Bei fehlender Angabe werden alle Sätze gelistet.
LISTING ACCESS NAME:	O	Der Gerätename eines Gerätes oder der Pfadname einer Datei, wohin die Inhalte der Sätze von TAXO geschrieben werden sollen. Bei fehlender Angabe wird der dem Terminal zugehörige Terminal Local File (TLF) als Ausgabedatei verwendet.
FOREGROUND (Y/N):	R	Die Eingabe von Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N(NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

2.6.3 Beispiel zum LLR Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein LLR Kommando an seinem Terminal eingeben kann, um die ersten drei Sätze einer Datei ausgeben zu lassen. Das Wort "SAME" in diesem Beispiel zeigt an, daß das letzte gedruckte Wort in der vorhergehenden Zeile das gleiche wie alle folgenden Wörter ist, bis das Ende eines Satzes erreicht ist oder ein unterschiedliches Wort kommt. Das letzte Wort eines Satzes wird immer gedruckt.

```
( ) LLR
LIST LOGICAL RECORD
 PATHNAME: .LLREG
 STARTING RECORD: 0
 NUMBER OF RECORDS: 3
 LISTING ACCESS NAME:                    HEXADEZIMAL        ASCII DARSTELLUNG
 FOREGROUND (Y/N): YES                   DARSTELLUNG        DES INHALTES
FILE ACCESS NAME: .LLREG                 DES INHALTES
RECORD: 000000
                                                  |                  |
0000  5245 434F 5244 204E 554D 4245 5220 4F4E     RE CO RD  N UM BE R  ON
0010  4520 2020 2020 2020 2020 2020 2020 2020     E
  SAME
004E  2020 ——   Das letzte Wort eines logischen Satzes wird immer ausgedruckt.
RECORD:  000001
0000  5245 434F 5244 204E 554D 4245 5220 5457     RE CO RD  N UM BE R  TW
0010  4F20 2020 2020 2020 2020 2020 2020 2020     O
  SAME
004E  2020
RECORD:  000002
0000  5245 434F 5244 204E 554D 4245 5220 5448     RE CO RD  N UM BE R  TH
0010  5245 4520 2020 2020 2020 2020 2020 2020     RE E
  SAME
004E  2020
[]
```

2.7 Ausgabe von Dateiinhalten (SF-Kommando)

Der Benutzer gibt das SF (SHOW-FILE) Kommando ein, um Inhalte einer Datei an seinem VDT anzuzeigen. Anwender mit Bildschirm-Terminals im VDT-Modus sehen nur den ersten Teil der Datei. Um den Rest der Datei auf dem Bildschirm anzeigen zu können, muß der Benutzer bestimmte Funktionstasten der Tastatur drücken, wie sie in der Tabelle 2-4 beschrieben sind. Während die Inhalte einer Datei angezeigt werden, kann der Benutzer kein anderes Kommando eingeben. Durch das SF-Kommando können sequentielle, relative und Key Index Dateien angezeigt werden.
Key Index Dateien können in der Sortierreihenfolge des Primärschlüssels angezeigt werden, wenn sie unter TAXO v 2.0.0 nach dem "Sequential Placement" Verfahren angelegt wurden.

Tabelle 2-4 Anzeigen einer Datei durch Kontrolltasten (nur im VDT-Modus)

Taste(n)	Funktion
F1	Wird verwendet, um den nächsten Teil einer Datei anzuzeigen. Wenn das End-of-File angezeigt wird, ist weiteres Drücken der F1-Taste wirkungslos.
F2	Wird verwendet, um einen vorangegangenen Teil einer Datei anzuzeigen. Wird der Beginn einer Datei angezeigt, ist weiteres Drücken der F2-Taste wirkungslos.
UP ARROW	Wird verwendet, um die nächste Zeile einer Datei in der letzten Zeile des Bildschirms anzuzeigen, die erste angezeigte Zeile verschwindet, alle anderen Zeilen werden jeweils eine Zeile höher auf dem Bildschirm angezeigt.
DOWN ARROW	Wird verwendet, um die nächste Zeile einer Datei in der ersten Zeile des Bildschirms anzuzeigen. Die letzte Zeile verschwindet vom Bildschirm, alle anderen Zeilen werden jeweils eine Zeile tiefer auf dem Bildschirm angezeigt.
+ integer und RETURN	Die Eingabe eines Plus-Zeichens, einer Ganzzahl (dezimal oder hexadezimal) und anschließendes Drücken der RETURN-Taste positioniert die Anzeige des Dateiinhaltes um die durch die Ganzzahl spezifizierte Zahl von Zeilen nach hinten (Richtung Dateiende).
- integer und RETURN	Die Eingabe eines Minus-Zeichens, einer Ganzzahl (dezimal oder hexadezimal) und anschließendes Drücken der RETURN-Taste positioniert die Anzeige des Dateiinhaltes um die durch die Ganzzahl spezifizierte Zahl von Zeilen nach vorne (Richtung Dateianfang).
integer und RETURN	Die Eingabe einer vorzeichenlosen Ganzzahl (dezimal oder hexadezimal) und ein anschließendes Drücken der RETURN-Taste positioniert die Anzeige der Dateiinhalte auf die Zeile, die durch die Ganzzahl angegeben ist.
COMMAND	Schließt das Anzeigen einer Datei ab und erlaubt die Eingabe eines anderen Kommandos.

2.7.1 Format des SF-Kommandos

SF
SHOW FILE
FILE PATHNAME: <acnm> @

2.7.2 Antworten des Benutzers beim SF-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antwort des Benutzers
FILE PATHNAME	R	Der Pfadname, der die Datei, die angezeigt werden soll, bezeichnet.

2.7.3 Beispiel zum SF-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das SF-Kommando eingeben kann, um Inhalte einer Datei an einem Datenterminal auszugeben.

```
( ) SF
    FILE PATHNAME: VOL 1.DIRA.FILE1
```

2.8 Ausgeben von Dateiinhalten auf einem Drucker - der TAXO Spooler

Mit Hilfe des PF (Print File) Kommandos ist es möglich, den Inhalt einer Datei oder einer Liste von Dateien auf einem Drucker auszugeben. Dieses Kommando erzeugt auf Grund der vom Benutzer spezifizierten Angaben einen oder mehrere Druckaufträge an den TAXO Spooler. Der TAXO Spooler ist eine Dienstleistung, die beim Laden des TAXO Betriebssystems automatisch initialisiert wird. Der TAXO Spooler sammelt alle Druckaufträge, die von beliebigen Terminals oder Hintergrundprozessen (auch gleichzeitig) erzeugt werden können, und reiht sie in eine Warteschlange (Queue) ein. Jedem Druckauftrag wird eine eindeutige Spool ID zugewiesen, die es erlaubt, den Druckauftrag bei Statusabfragen und Fehlermeldungen zu identifizieren oder ihn wieder aus der Warteschlange zu entfernen.

Jeder Druckauftrag in der Warteschlange enthält folgende Angaben:

- die Spool ID
- das Gerät, auf dem ausgedruckt werden soll
- eine Formularnummer, durch die es möglich ist, eine Datei erst auszudrucken, wenn ein bestimmtes Formular in den Drucker eingelegt ist
- die Priorität, die angibt, ob eine Datei bevorzugt vor anderen ausgedruckt werden soll

Stellt der Spooler fest, daß ein Druckauftrag auf einem bestimmten Drucker beendet ist, sucht er in der Warteschlange nach einem weiteren Auftrag, der auf diesem Gerät abgearbeitet werden soll, bei dem die angegebene Formularnummer mit der am Drucker eingestellten übereinstimmt und der bevorzugt ausgedruckt werden soll. Ist kein Auftrag mit bevorzugtem Druck vorhanden, wird der zeitlich erste Auftrag mit passender Formularnummer und normaler Priorität ausgewählt. Ist kein Auftrag mit passender Formularnummer vorhanden, so wartet der Spooler, bis ein neuer passender Auftrag für das Gerät eintrifft oder von einem beliebigen Benutzer ein Formularwechsel durchgeführt wird.

Das Wechseln des Formulars ist von einem beliebigen Terminal mit Hilfe des MSD (Modify Spooler Device) Kommandos möglich. Um einen Formularwechsel auf einem Drucker durchführen zu können, muß der gewünschte Drucker frei sein, d.h. es dürfen keine passenden Aufträge mehr in der Spooler Auftragswarteschlange vorhanden sein oder die Ausgabeaktivitäten wurden mit dem HO (Halt Output) Kommando am Ende des letzten Auftrags gestoppt (Halt Output at EOF). Nach erfolgtem Formularwechsel wird vom Spooler entweder automatisch oder - falls das Drucken mit HO unterbrochen wurde - nach Eingabe des RO (Resume Output) Kommandos der nächste passende Auftrag ausgewählt.

Während des Formularwechsels ist es möglich, eine vom Benutzer vorgegebene Datei beliebig oft auszudrucken, um die richtige Justage des neuen Formulars feststellen zu können.

Einer Formularnummer können außerdem spezielle Funktionen eines Druckers zugeordnet werden. Diese speziellen Funktionen können durch "Code Sequenzen" in einer "Formularsteuerdatei" festgelegt werden. Sie sind abhängig vom verwendeten Drucker und können z. B. sein: Drucken mit alternativem Zeichenvorrat, Drucken in einer anderen Schriftart u. ä.

Neben den bisher beschriebenen Funktionen stehen noch folgende Dienstleistungen des Spoolers zur Verfügung:

- Erstellen einer temporären Datei, die nur einen gewünschten Ausschnitt (Angabe von Start- und Endeseite!) aus der ursprünglichen Datei enthält. Dabei ist es möglich, eine beliebige Textzeile auf jeder Seite einzufügen, um z.B. gesetzliche Vorschriften über die Kennzeichnung von Zweitdrucken zu erfüllen.

- Automatisches Löschen einer Druckdatei nach erfolgtem Ausdrucken.

- Abbrechen des laufenden Auftrags auf einem Drucker oder Entfernen einzelner oder aller Druckaufträge für einen Drucker aus der Warteschlange (KO - Kill Output-Kommando).

- Mitführen eines Protokolls ("History File") über (alle) abgesetzten Druckaufträge.

- Anzeigen des Druckprotokolls (S$PF - Show Printed Files)

- Löschen der im History File enthaltenen Dateien nach erfolgreichem Ausdruck an Hand eines Löschkennzeichens (D$PF - Delete Printed Files-Kommando).

- Anzeigen der an den Druckern eingestellten Formularnummer und der Spooler- Warteschlange (SOS - Show Output Status - Kommando).

- Einstellen der Druckbreite (LPWIDTH - Kommando).

- Aufbereiten einer Datei zum Ausdrucken auf einem Drucker mit einem speziellen (landesspezifischen) Zeichenvorrats (IPF - International Print File - Kommando).

In den folgenden Abschnitten 2.8.1 bis 2.8.9 werden die einzelnen Kommandos genau beschrieben.

2.8.1 Erstellen von Druckaufträgen - PF (Print File) Kommando

Mit Hilfe des PF (Print File) Kommandos kann der Benutzer die für einen Druckauftrag notwendigen Angaben an den Spooler schicken. Der Spooler weist dem Druckauftrag eine eindeutige Spool ID der Form SXXXXX (mit XXXXX = lfd. Auftragsnummer) zu. Diese Spool ID wird bei späteren Status- und Fehlermeldungen für den Druckauftrag mit ausgegeben. Außerdem kann der Druckauftrag durch Angabe seiner Spool ID mit dem KO (Kill Output) Kommando aus der Spooler Warteschlange wieder entfernt werden. Es können sequentielle und relative Dateien ausgedruckt werden. Enthält die Datei im ersten Satz Druckersteuerzeichen (Zeilenvorschub, Blattwechsel u.ä.), wird die Datei so ausgedruckt, wie es die vorhandenen Steuerzeichen vorschreiben. Enthält die Datei im ersten Satz keine Steuerzeichen, wird vom Spooler automatisch nach jedem Satz ein Zeilenvorschub erzeugt. Enthält die Datei zusätzlich ANSI-Steuerzeichen (siehe Tabelle 2.5) und wurde ANSI FORMAT YES angegeben, so wird die entsprechende Funktion ausgeführt. Die ANSI-Steuerzeichen selbst werden dann nicht ausgegeben.

ANSI-Steuerzeichen müssen an der ersten Stelle des Satzes stehen, vor dem der Drucker die gewünschte Funktion ausführen soll. ANSI-Steuerzeichen an einer anderen Position innerhalb eines Satzes werden als normale ASCII-Zeichen behandelt.

Tabelle 2.5: ANSI-Steuerzeichen und ihre Bedeutung

blank	Eine Zeilenschaltung vor Druck des Satzes
0	Zwei Zeilenschaltungen vor Druck des Satzes
1	Vorschub auf nächsten Seitenanfang vor Druck des Satzes
+	Keine Zeilenschaltung vor Druck des Satzes (Überschreiben)
#	Sonderfunktion für bestimmte Drucker vor Druck des Satzes (z.B. gesperrte Schrift u. ä.; das # Zeichen bewirkt die Ausgabe des ASCII Zeichens >0E an den Drucker, wodurch – in Abhängigkeit vom verwendeten Drucker – in der Hardware vorhandene Sonderfunktionen eingestellt werden können.

ANMERKUNG:

Die logische Satzlänge, die beim Kreieren der auszudruckenden Datei festgelegt wurde, legt die Zahl der Zeichen fest, die maximal in einer Zeile gedruckt werden. Längere Sätze werden abgeschnitten. Alle Sätze, die länger als die vom Drucker verarbeitbare Zeilenlänge sind, werden ebenfalls abgeschnitten.

2.8.1.1 Format des PF-Kommandos

```
PF
PRINT FILE
FILE PATHNAME(S): (<acnm>@)
ANSI FORMAT?: <yes/no>
LISTING DEVICE: <acnm>@
DELETE AFTER PRINTING?: [ string ]
NUMBER OF LINES/PAGE: [ int ]
FORM: [ int ]
PREFERRED PRINTING? [ yes/no ]
NUMBER OF COPIES: [ int ]
SHOW OUTPUT STATUS: [ yes/no ]
```

2.8.1.2 Benutzerantworten beim PF-Kommando

FILE PATHNAME	R	Ein Pfadname oder eine Liste von Pfadnamen, die die auszudruckende(n) Datei(en) angibt.
ANSI FORMAT	R	Y (yes) besagt, daß die zu druckende(n) Datei(en) ANSI-Steuerzeichen enthält (enthalten). N(no) besagt, daß keine ANSI Steuerzeichen vorhanden sind.
LISTING DEVICE	R	Ein Gerätename, der das Gerät, auf dem gedruckt werden soll, bezeichnet.

DELETE AFTER PRINTING	O	Hier kann der Benutzer durch Eingabe eines beliebigen Zeichens angeben, ob die Datei nach dem Ausdrucken automatisch gelöscht werden soll (Eingabe von Y), ob sie erhalten bleiben soll (keine Eingabe oder Eingabe von N) oder ob sie zu einem späteren Zeitpunkt durch das D$PF-Kommando gelöscht werden soll (Eingabe eines beliebigen anderen Zeichens)
NUMBER OF LINES/PAGE	O	Anzahl der zu druckenden Zeilen pro Seite. Standardwert = 62
FORM	O	Die Nummer des Formulars, auf dem die Datei gedruckt werden soll. Die Formularnummer muß im Bereich 0 (kein spezielles Formular) bis 32767 liegen. Die Datei wird erst gedruckt, wenn am Drucker das entsprechende Formular eingestellt ist. Wird keine Formularnummer angegeben, so wird FORM = 0 angenommen. <u>Anmerkungen:</u> Die Angabe einer Formularnummer hat nur dann eine Wirkung, wenn für LISTING DEVICE LP01...LP16 (Systemdrucker) oder SP01...SP40 (Arbeitsplatzdrucker) angegeben wurde. Bei anderen Gerätenamen wird die angegebene Formularnummer ignoriert. Existiert für die angegebene Formularnummer eine Formularsteuerdatei, so wird deren Inhalt beim Ausdrucken berücksichtigt. Für Druckaufträge ohne Formularnummer wird die Formularsteuerdatei für Formularnummer 0 verwendet (soweit vorhanden).
PREFERRED PRINTING?	O	Durch Eingabe von Y (yes) kann erreicht werden, daß die Datei bevorzugt ausgedruckt wird, d.h. daß sie vor allen anderen Dateien mit gleicher Formularnummer, aber nichtbevorzugtem Druck, ausgegeben wird. Die Eingabe von N (no) oder keine Eingabe bewirkt, daß die Datei nicht bevorzugt ausgedruckt wird. <u>Anmerkung</u> Die Eingabe von Y (yes) hat nur dann eine Wirkung, wenn gleichzeitig eine Formularnummer angegeben wird. Ist keine Formularnummer angegeben, so wird die hier gemachte Eingabe ignoriert.
NO OF COPIES:	O	Hier kann der Benutzer festlegen, wie oft die angegebene Datei bzw. Dateiliste ausgedruckt werden soll. Keine Angabe bedeutet ein Exemplar.
SHOW OUTPUT STATUS:	O	Bei Angabe von Y (yes) wird vor Beendigung des PF-Kommandos der Spooler-Status für das angegebene LISTING DEVICE angezeigt. Bei anderen Angaben wird der Status nicht angezeigt.

2.8.1.3 Beispiel für ein PF-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer eine Datei auf dem Systemdrucker LP01 viermal ausdrucken kann.

Der Name der Datei ist VOL1.MYFILE. Die Datei soll auf normalem Papier gedruckt werden (d.h. kein spezielles Format).

```
( ) PF
    PRINT FILE
    FILE PATHNAME(S): VOL1.MYFILE
    ANSI FORMAT?: NO
    LISTING DEVICE: LP01
    DELETE AFTER PRINTING: NO
    NUMBER OF LINES/PAGE:
    FORM:
    PREFERRED PRINTING?:
    NUMBER OF COPIES: 4
    SHOW OUTPUT STATUS: NO
( )
```

2.8.2 Druckunterbrechung (HO-Kommando)

Benutzer können das HO-Kommando (HALT OUTPUT) eingeben, um vorübergehend die Ausgabe für das angegebene Gerät anzuhalten (z.B. zum Wechseln des Formulars auf dem Zeilendrucker). Das HO-Kommando kann vom Benutzer immer dann eingegeben werden, wenn der Druck angehalten werden soll. Der Benutzer gibt ein "I" ein, wenn der Stop des Ausdrucks sofort und ein "E", wenn der Stop erst am Ende der Datei (EOF) gewünscht wird. Standardwert ist "I" - sofortiger Stop.

Der Benutzer hebt die Druckunterbrechung durch Eingabe des RO-Kommandos (RESUME OUTPUT) auf. Bei Eingabe des RO-Kommandos wird der Ausdruck an der Stelle fortgeführt, an der unterbrochen wurde.

2.8.2.1 Format des HO-Kommandos

HO
HALT OUTPUT AT DEVICE
DEVICE NAME: <acnm> @
IMMEDIATELY OR AT EOF? (I/E): [string]

2.8.2.2 Antworten des Benutzers beim HO-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
DEVICE NAME:	R	Der Gerätename des Gerätes, auf dem der Benutzer die Ausgabe unterbrechen will.
IMMEDIATELY OR AT EOF (I/E):	O	I muß eingegeben werden, wenn der Benutzer einen sofortigen Stop des Ausdrucks wünscht. E ist erforderlich, wenn der Stop am Ende der Datei erfolgen soll. Standardwert ist ein sofortiger Stop (I). Jedes andere eingegebene Zeichen als I wird als E interpretiert.

2.8.2.3 Beispiel für ein HO-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein HO-Kommando eingeben kann, um eine zeitweise Druckunterbrechung auf dem Zeilendrucker zu bewirken.

```
( ) HO
HALT OUTPUT AT DEVICE
  DEVICE NAME: LP01
  IMMEDIATELY OR AT EOF? (I/E): I
( )
```

2.8.3 Fortsetzen der Ausgabe (RO-Kommando)

Man verwendet das RO-Kommando (RESUME OUTPUT), um eine Druckunterbrechung aufzuheben und die Ausgabe auf dem Gerät fortzusetzen. Nach der Eingabe eines RO-Kommandos wird die Ausgabe an dem Punkt fortgesetzt, wo der Druck unterbrochen wurde.

2.8.3.1 Format des RO-Kommandos

RO
RESUME OUTPUT AT DEVICE
DEVICE NAME: <acnm> @

2.8.3.2 Antworten des Benutzers beim RO-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antwort des Benutzers
DEVICE NAME:	R	Der Gerätename des Gerätes, auf dem die Ausgabe fortgesetzt werden soll.

2.8.3.3 Beispiel für ein RO-Kommando

Das folgende Beispiel zeigt, wie man ein RO-Kommando eingeben kann, um eine Ausgabe, die zeitweise unterbrochen wurde, auf einem Zeilendrucker fortzusetzen.

```
( ) RO
    RESUME OUTPUT AT DEVICE
    DEVICE NAME: LP01
( )
```

2.8.4 Beenden einer Ausgabe (KO-Kommando)

Man verwendet das KO-Kommando (KILL OUTPUT), um einen oder alle Druckaufträge auf einem Gerät abzubrechen.

Anmerkung:
Bei KO werden alle Aufträge für den jeweiligen Drucker gelöscht und nicht nur die der jeweiligen User-ID.

2.8.4.1 Format des KO-Kommandos

KO
DEVICE NAME OR SPOOL ID: <acnm> @
KILL CURRENT FILE ONLY?: [yes/no]

2.8.4.2 Antworten des Benutzers beim KO-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
DEVICE NAME OR SPOOL ID	R	Der Gerätename des Gerätes, auf dem die Ausgabe beendet werden soll oder die Spool ID eines Druckauftrags der aus der Warteschlange entfernt werden soll. Die Angabe beim 2. Parameter hat dann keine Bedeutung.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
KILL CURRENT FILE ONLY?:	O	Y (yes) beendet die Ausgabe der ersten Datei, die in der Warteschlange eingetragen ist. N (no) beendet die Ausgabe aller Dateien, die in der Warteschlange stehen: es erfolgt genau eine Meldung über die Systemzeile an die Terminals, von denen die Druckaufträge stammen. Die Beendigung der ersten Datei, die in der Warteschlange steht, ist Standard. Wenn alle Dateien beendet werden (no), so muß der Benutzer das RO-Kommando eingeben, um das Ausgabegerät zu aktivieren; ansonsten bleibt es inaktiv. <u>Anmerkung:</u> Wird beim 1. Parameter eine Spool ID angegeben, so hat eine hier getätigte Eingabe keine Bedeutung.

2.8.4.3 Beispiel für ein KO-Kommando

Das folgende Beispiel zeigt, wie man ein KO-Kommando eingeben kann, um eine Ausgabe auf einem Zeilendrucker zu beenden.

```
( ) KO
    KILL OUTPUT AT DEVICE
    DEVICE NAME OR SPOOL ID : LP01
    KILL CURRENT FILE ONLY?: YES
( )
```

2.8.5 Anzeigen des Spooler Status (SOS Kommando)

Durch Aufruf des SOS (Show Output Status) Kommando wird der aktuelle Status des Spooler Systems angezeigt. Die Statusanzeige enthält, in Abhängigkeit von den SOS Aufrufparametern, folgende Informationen:

- Die an einem angegebenen Drucker eingestellte Formularnummer
- Alle an den verfügbaren Druckern eingestellten Formularnummern
- Alle Druckaufträge eines angegebenen Benutzers, die an einem angegebenen Drucker anstehen.
- Alle Druckaufträge eines angegebenen Benutzers, die zur Abarbeitung anstehen.
- Alle Druckaufträge, die für einen angegebenen Drucker anstehen.
- Alle Druckaufträge, die zur Abarbeitung in der Spooler Warteschlange stehen.

Die Abb. 2.7 zeigt das Format der durch einen SOS Befehl erzeugten Ausgabe.
Die Tabelle 2.6 beschreibt die einzelnen Informationen der Statusanzeige.

```
FORM LIST

 DEVICE    FORM
  LP02        5
  SP02        7
END OF FORM LIST

PRINT QUEUE

  ST/  USER  DEVICE    FORM  DEL  SPOOL ID  FILE NAME
  PRI   ID
  W/1 SYS000  LP01      10   N    S00001   .TA$PROC.IS
  A/1 SYS000  LP01           N    S00002   .TA$PROC.SF
  A/1 SYS000  SP02       7   N    S00003   .TA$PROC.MD
  W/1 SYS000  LP02           N    S00004   .TA$PROC.FCU
END OF PRINT QUEUE
```

Abb. 2.7 Spooler Statusanzeige durch SOS

Die Anzeige besteht aus 2 Teilen:

Die FORM LIST (Formularliste) enthält die an den Druckern eingestellten Formularnummern. Dabei steht in der Spalte DEVICE der Gerätename des Druckers und in der Spalte FORM die eingestellte Formularnummer.
Die Formularliste wird durch die Zeile END OF FORM LIST beendet.
Die FORM LIST enthält nur Angaben für Drucker, an denen ein spezielles Formular eingestellt ist.

Die PRINT QUEUE (Warteschlange der Druckaufträge) enthält Angaben über die anstehenden Druckaufträge. Die Bedeutung der einzelnen Spalten:

Überschrift	Bedeutung
ST/PRI	Vor dem Schrägstrich steht der Status des Druckauftrages: A = aktiv, d.h. wird gerade gedruckt W = wartet, d.h. wird noch nicht gedruckt Hinter dem Schrägstrich steht die Priorität des Druckauftrages: 0 = bevorzugter Ausdruck 1 = kein bevorzugter Ausdruck
USER ID	Benutzerkennung, von der der Druckauftrag erzeugt wurde.
DEVICE	Angabe des Gerätes, für das der Auftrag gilt.
SPOOL ID	Die dem Druckauftrag zugewiesene Spool ID
FORM	Angabe der Formularnummer, die dem Druckauftrag mitgegeben wurde.
DEL	Angabe, ob die Datei nach Ausdruck automatisch (Anzeige = Y) oder nicht automatisch (Anzeige = N) gelöscht werden soll. Die Anzeige N erscheint für alle Druckaufträge, bei denen ein anderes Löschkennzeichen als Y beim PF Kommando spezifiziert wurde.
FILE NAME	Pfadname der auszudruckenden Datei

Die Warteschlange der Druckaufträge wird durch die Zeile END OF PRINT QUEUE abgeschlossen.

2.8.5.1 Format des SOS-Kommandos

SOS
SHOW OUTPUT STATUS
USER ID: [name]
DEVICE NAME: [name]

2.8.5.2 Benutzerantworten zum SOS Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzereingaben
USER ID	O	Wird eine Benutzerkennung (USER ID) angegeben, so wird nur der Status der Druckaufträge für diese Benutzerkennung angezeigt. Wird keine Eingabe gemacht, so wird der Status aller Druckaufträge angezeigt.
DEVICE NAME	O	Wird hier der Name eines Druckers angegeben, (z.B. LP01 oder SP03), so werden nur die an diesem Drucker eingestellte Formularnummer und die für diesen Drucker anstehenden Druckaufträge angezeigt. Wird keine Eingabe gemacht, so werden diese Informationen für alle Drucker angezeigt.

2.8.5.3 Beispiel zum SOS Kommando

Im folgenden Beispiel setzen 3 Benutzer je 2 Druckaufträge für verschiedene Drucker ab. Nachdem alle Aufträge dem Spooler übergeben sind, werden verschiedene SOS Aufrufe durchgeführt.

Benutzer SYS001:

```
() PF
  PRINT FILE
  FILE PATHNAME(S): .TA*PROC.LO
  ANSI FORMAT?: NO
  LISTING DEVICE: LP02
  DELETE AFTER PRINTING?: NO
  NUMBER OF LINES/PAGE:
  FORM:
  PREFERRED PRINTING?: NO
  NUMBER OF COPIES: 1
  SHOW OUTPUT STATUS: NO

() PF
  PRINT FILE
  FILE PATHNAME(S): .TA*PROC.MO
  ANSI FORMAT?: NO
  LISTING DEVICE: LP01
  DELETE AFTER PRINTING?: NO
  NUMBER OF LINES/PAGE:
  FORM:
  PREFERRED PRINTING?: NO
  NUMBER OF COPIES: 1
  SHOW OUTPUT STATUS: NO
```

Benutzer SYS002:

```
() PF
  PRINT FILE
  FILE PATHNAME(S): .TA*PROC.XB
  ANSI FORMAT?: NO
  LISTING DEVICE: LP03
  DELETE AFTER PRINTING?: NO
  NUMBER OF LINES/PAGE:
  FORM:
  PREFERRED PRINTING?: NO
  NUMBER OF COPIES: 1
  SHOW OUTPUT STATUS: NO

() PF
  PRINT FILE
  FILE PATHNAME(S): .TA*PROC.DO
  ANSI FORMAT?: NO
  LISTING DEVICE: LP04
  DELETE AFTER PRINTING?: NO
  NUMBER OF LINES/PAGE:
  FORM:
  PREFERRED PRINTING?: NO
  NUMBER OF COPIES: 1
  SHOW OUTPUT STATUS: NO
```

Benutzer SYS003:

```
() PF
   PRINT FILE
   FILE PATHNAME (S): .TA$PROC.MVI
   ANSI FORMAT?: NO
   LISTING DEVICE: LP02
   DELETE AFTER PRINTING?: NO
   NUMBER OF LINES/PAGE:
   FORM:
   PREFERRED PRINTING?: NO
   NUMBER OF COPIES: 1
   SHOW OUTPUT STATUS: NO

() PF
   PRINT FILE
   FILE PATHNAME (S): .TA$PROC.CKS
   ANSI FORMAT?: NO
   LISTING DEVICE: LP01
   DELETE AFTER PRINTING?: NO
   NUMBER OF LINES/PAGE:
   FORM:
   PREFERRED PRINTING?: NO
   NUMBER OF COPIES: 1
   SHOW OUTPUT STATUS: NO

() SOS
   SHOW OUTPUT STATUS
      USER ID:
      DEVICE NAME:

FORM LIST

 DEVICE FORM
END OF FORM LIST

PRINT QUEUE

 ST/   USER    DEVICE   FORM   DEL   SPOOL ID   FILE NAME
 PRI    ID
 A/1  SYS001   LP02            N     S00001     .TA$PROC.LD
 A/1  SYS001   LP01            N     S00002     .TA$PROC.MD
 A/1  SYS002   LP03            N     S00003     .TA$PROC.XB
 A/1  SYS002   LP04            N     S00004     .TA$PROC.OD
 W/1  SYS003   LP02            N     S00005     .TA$PROC.MVI
 W/1  SYS003   LP01            N     S00006     .TA$PROC.CKS
END OF PRINT QUEUE

() SOS
   SHOW OUTPUT STATUS
      USER ID: SYS001
      DEVICE NAME:

FORM LIST

 DEVICE FORM
END OF FORM LIST
```

```
PRINT QUEUE

  ST/  USER    DEVICE   FORM   DEL   SPOOL ID   FILE NAME
  PRI   ID
  A/1 SYS001    LP02            N     S00001    .TA$PROC.LD
  A/1 SYS001    LP01            N     S00002    .TA$PROC.MD
END OF PRINT QUEUE

( )  SOS
     SHOW OUTPUT STATUS
        USER ID:
        DEVICE NAME: LP02


FORM LIST

 DEVICE FORM
END OF FORM LIST



PRINT QUEUE

  ST/  USER  DEVICE  FORM  DEL  SPOOL ID  FILE NAME
  PRI   ID
  A/1 SYS001  LP02          N    S00001   .TA$PROC.LD
  W/1 SYS003  LP02          N    S00002   .TA$PROC.MVI
END OF PRINT QUEUE
```

2.8.6 Formularwechsel - das MSD-Modify Spooler Device Kommando

Sollen Dateien auf einem speziellen Formular ausgedruckt werden, (z.B. Schecks, Rechnungen u. ä.), so kann beim Absetzen des Druckauftrages (PF - Print File Kommando) eine Formularnummer angegeben werden. Die Datei wird dann erst gedruckt, wenn auf dem Drucker diese Formularnummer eingestellt ist. Einer Formularnummer können außerdem spezielle Druckerfunktionen in einer Formularsteuerdatei zugeordnet werden (z. B. Formularlänge, Schriftart u. a.), die beim Einstellen der Formularnummer automatisch über Code Sequenzen ausgewählt werden (druckerabhängig!). Das Einstellen der Formularnummer geschieht mit dem MSD (Modify Spooler Device) Kommando. Zum Wechseln des Formulars ist es notwendig, daß der entsprechende Drucker frei ist, d.h. es darf keine Druckausgabe mehr anstehen. Das ist der Fall, wenn alle Aufträge für diesen Drucker mit der aktuellen Formularnummer abgearbeitet sind oder wenn eine Druckunterbrechung am Ende einer Datei mit HO (Halt Output at EOF) erzeugt wurde. Nun kann das MSD-Kommando mit der neuen Formularnummer eingegeben werden. Dabei ist es auch möglich, daß beliebig oft ein Probedruck gemacht wird, um die richtige Justage des neuen Formulars überprüfen zu können. Das MSD-Kommando prüft zunächst, ob die Bedingungen für einen Formularwechsel erfüllt sind (siehe oben). Ist dies der Fall, wird der Benutzer durch eine entsprechende Meldung zum Wechseln des Formulars aufgefordert. Nach dem Formularwechsel werden in einer Formularsteuerdatei eventuell definierte Code Sequenzen an den Drucker ausgegeben. Soll nun ein Probedruck durchgeführt werden, wird die angegebene Datei einmal ausgedruckt. Nun kann der Benutzer durch Beantworten einer Meldung die richtige Justage bestätigen oder die Datei für den Probedruck nochmals drucken lassen. Ist das MSD-Kommando beendet, wird vom Spooler der nächste passende Auftrag ausgedruckt.

ANMERKUNG:

Während eines Formularwechsels ist für den Drucker die Formularnummer 32768 eingestellt. Diese Formularnummer kann von einem Benutzer nicht eingegeben werden, sie wird jedoch in der SOS-Ausgabe angezeigt.

2.8.6.1 Format des MSD-Kommandos

```
MSD
MODIFY SPOOLER DEVICE
DEVICE NAME: <name> @
FORM: [ int ]
PRINT TEST SHEET?(Y/N): [ yes/no ]
TEST SHEET FILE PATHNAME: [ acnm ]
```

2.8.6.2 Benutzerantworten zum MSD-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
DEVICE NAME	R	Gerätename des Druckers, auf dem ein Formularwechsel durchgeführt werden soll.
FORM	O	Angabe der neuen Formularnummer. Sie muß zwischen 0 und 32767 liegen. Wird keine Angabe gemacht, so bedeutet dies: Formularnummer 0.

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
PRINT TEST SHEET? (Y/N)	O	Eingabe von Y(yes) bedeutet, daß nach dem Einlegen des neuen Formulars ein Probedruck durchgeführt werden soll. Eingabe von N(no) oder keine Eingabe bedeutet, keinen Probedruck.
TEST SHEET FILE PATH-NAME	O	Hier kann der Name einer Datei angegeben werden, die für den Probedruck verwendet werden soll. Wird keine Angabe gemacht, so wird für den Probedruck eine Testdatei verwendet. Der Pfadname der Testdatei ist .S$PRINT.TEST. Diese Testdatei kann vom Benutzer selbst erstellt werden.

2.8.6.3 Dialogmeldungen des MSD-Kommandos

Nach Eintasten aller Aufrufparameter prüft das MSD-Kommando, ob auf dem angegebenen Gerät ein Formularwechsel möglich ist. Ist das Gerät nicht verfügbar, weil es z.B. von einem anderen Programm belegt ist, wird die Meldung

DEVICE NOT AVAILABLE. ERROR CODE = 0XXXX.

ausgegeben. Dabei ist XXXX die Fehlernummer, die angibt, warum das Gerät nicht verfügbar ist. Sie kann im TAXO Fehlerhandbuch im Kap. 6 nachgeschlagen werden.
Handelt es sich bei dem angegebenen Gerät nicht um einen Drucker, so wird die Meldung

DEVICE NOT A PRINTER

ausgegeben. Jede dieser Meldungen muß vom Benutzer durch Drücken der Return Taste quittiert werden. Das Kommando wird dann abgebrochen.

Kann ein Formularwechsel durchgeführt werden, so wird der Benutzer mit der Meldung

MOUNT FORM n ON DDDD

TYPE "C" TO CONTINUE, "$" TO ABORT:

aufgefordert, das neue Formular einzulegen. Dabei bezeichnet n die vom Benutzer angegebene Formularnummer und DDDD das vom Benutzer angegebene Gerät. Der Benutzer kann an dieser Stelle das MSD-Kommando durch Eingabe eines $-Zeichens abbrechen. Es ist jedoch zu beachten, daß dann die Formularnummer 32768 (für den Benutzer nicht erlaubt) eingestellt ist. Deshalb muß das MSD-Kommando nochmals aufgerufen werden. Eine gültige Formularnummer ist erst nach kompletter Durchführung von MSD wieder eingestellt.

Ist das richtige Formular eingelegt und justiert, kann durch Eingabe des Buchstabens C das MSD-Kommando fortgesetzt werden.

Wird auf diese Meldung ein anderes Zeichen als $ bzw. C eingegeben, wird die Meldung wiederholt.

Jede Eingabe auf diese Meldung muß mit der Return Taste abgeschlossen werden.

Wurde beim MSD Aufruf ein Probedruck verlangt, wird ein Druckauftrag mit Formularnummer 32768 für die Probedruckdatei erzeugt und abgearbeitet. Es erscheint nun die Meldung

CHECK TEST SHEET ON DDDD.

TYPE "C" TO CONTINUE, "R" TO REPEAT:

Der Benutzer wird damit aufgefordert, den Probedruck auf dem Gerät DDDD zu überprüfen. Durch Eingabe des Buchstabens R kann der Probedruck beliebig oft wiederholt werden. Durch Eingabe des Buchstabens C wird das MSD-Kommando beendet. Die neue Formularnummer ist für das angegebene Gerät eingestellt. Alle Eingaben auf die Meldung müssen wieder durch die Return Taste abgeschlossen werden.

Wurde vor Aufruf des MSD-Kommandos der Ausdruck von Dateien mit HO (Halt Output) unterbrochen, muß er durch Eingabe des Kommandos RO (Resume Output) wieder gestartet werden.

2.8.6.4 Beispiele zum MSD-Kommando

Das folgende Beispiel 1 zeigt, wie auf dem Systemdrucker LP03 die Formularnummer 10 eingestellt werden kann, wenn noch mehrere Dateien für die gerade eingestellte Formularnummer vorhanden sind, der Wechsel jedoch so bald wie möglich erfolgen soll. Nach dem Formularwechsel soll ein Probedruck mit der Datei .MYTEST durchgeführt werden.

Beispiel 1:

```
() SOS
   DEVICE NAME: LP03

() HO
   DEVICE NAME: LP03
   IMMEDIATELY OR AT EOF? (I/E): E

() MSD
   FORM: 10
   DEVICE NOT AVAILABLE. ERROR CODE = 0000F.
```

Benutzer wartet einige Minuten.

```
() SOS
() MSD
MODIFY SPOOLER DEVICE
                DEVICE NAME:  LP03
                      FORM:  00010
   PRINT TEST SHEET?(Y/N):  YES
TEST SHEET FILE PATHNAME:  .MYTEST
() RO
```

Beispiel 2:

```
[] MSD

   MODIFY SPOOLER DEVICE

                DEVICE NAME:  SP02
                      FORM:  00005
   PRINT TEST SHEET? (Y/N):  YES
TEST SHEET FILE PATHNAME:  .TEST

     0123456789   0123456789  0123456789

     AAAAAAAAAA   AAAAAAAAAA  AAAAAAAAAA

     BBBBBBBBBB   BBBBBBBBBB  BBBBBBBBBB

     CCCCCCCCCC   CCCCCCCCCC  CCCCCCCCCC
```

Beispiel 3:

```
[] MSD

  MODIFY SPOOLER DEVICE

                DEVICE NAME:  SP02
                      FORM:  00100
   PRINT TEST SHEET? (Y/N):  YES
  TEST SHEET FILE PATHNAME:  .TEST


0123456789   0123456789   0123456789

AAAAAAAAAA   AAAAAAAAAA   AAAAAAAAAA

BBBBBBBBBB   BBBBBBBBBB   BBBBBBBBBB

CCCCCCCCCC   CCCCCCCCCC   CCCCCCCCCC
```

2.8.7 Erstellen eines Ausschnitts aus einer Druckdatei für Druckwiederholung – das BPPF (Build Partial Print File) Kommando

Vereinzelt ist es erforderlich, daß ein Teil einer Druckdatei noch einmal gedruckt werden soll (z.B. nach einem Farbband- oder Papierriß am Drucker). Für diesen Fall ist das BPPF (Build Partial Print File) Kommando auf die ursprüngliche Druckdatei anzuwenden. Das Kommando erstellt einen beliebigen Ausschnitt aus der ursprünglichen Druckdatei in eine vom Benutzer vorgegebene Zieldatei. Der Ausschnitt beginnt auf einer vom Benutzer vorgegebenen Startseite und endet auf einer vom Benutzer vorgegebenen Endeseite. Dabei ist es möglich, auf jeder Seite des erstellten Ausschnitts an einer beliebigen Stelle der Seite einen Text einzufügen. Enthält die ursprüngliche Seite an dieser Stelle bereits einen Text, so wird dieser durch den eingefügten Text überschrieben.

Durch diese Texteinfügungsmöglichkeit kann eine durch den Gesetzgeber vorgeschriebene Kennzeichnung von Zweitdrucken realisiert werden. Existiert die vom Benutzer angegebene Zieldatei bereits, wird sie vom BPPF-Kommando überschrieben. Ist sie noch nicht vorhanden, wird sie vom BPPF-Kommando automatisch angelegt.

Die mit BPPF erstellte Datei wird nicht automatisch ausgedruckt. Sie muß vielmehr durch einen eigenen Aufruf des PF- (Print File) Kommandos gedruckt werden. Hierdurch ist es möglich, den mit BPPF erstellten Ausschnitt vor dem Druck noch einmal zu überprüfen und den Druck mit den beim PF- (Print File) Kommando möglichen Parametern zu beeinflussen.

Die mit BPPF erstellten Ausschnittsdateien können im Directory .S$PRINT auf der Systemplatte angelegt und beim späteren Drucken mit PF- (Print File) automatisch gelöscht werden (DELETE AFTER PRINTING = YES).

2.8.7.1 Format des BPPF-Kommandos

```
( ) BPPF
    BUILD PARTIAL PRINT FILE
    ORIGINAL PRINT FILE PATHNAME: <acnm> @
    PARTIAL FILE PATHNAME: <acnm> @
    FIRST PAGE: [ int ]
    LAST PAGE: [ int ]
    ANSI FORMAT?: <yes/no>
    TEXT TO INSERT: [ string ]
    INSERT IN LINE: [ int ]
    COLUMN: [ int ]
    LINES PER PAGE: [ int ]
    FOREGROUND?(Y/N): <yes/no>
```

2.8.7.2 Benutzerantworten zum BPPF-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
ORIGINAL FILE PATHNAME	R	Der Pfadname der Druckdatei, von der ein Ausschnitt erstellt werden soll. Vorgabewert ist der Name der Datei, die beim letzten PF- (Print File) Aufruf angegeben wurde.
PARTIAL FILE PATHNAME	R	Der Pfadname der Zieldatei, in der der Ausschnitt erstellt werden soll. Dieser Name wird beim nächsten PF (Print File) als Vorgabewert angezeigt.
FIRST PAGE	O	Seitennummer aus der ursprünglichen Druckdatei, bei der der Ausschnitt beginnen soll. Standardwert = 1.
LAST PAGE	O	Seitennummer aus der ursprünglichen Druckdatei, bei der der Ausschnitt enden soll. Standardwert = 1024 bzw. Dateiende.
ANSI FORMAT	R	Y (yes) besagt, daß die zu drukkende(n) Datei(en) ANSI-Steuerzeichen enthält (enthalten). No (no) besagt, daß keine ANSI-Steuerzeichen vorhanden sind.
TEXT TO INSERT	O	Angabe eines beliebigen Textes, der auf jeder Seite des Ausschnitts eingefügt werden soll.
INSERT IN LINE	O	Angabe der Zeile, in der der bei TEXT TO INSERT angegebene Text eingefügt werden soll. Standardwert = 1.
COLUMN	O	Angabe der Spalte, ab der der bei TEXT TO INSERT angegebene Text innerhalb der bei INSERT IN LINE angegebenen Zeile eingefügt werden soll. Standardwert = 1.
LINES PER PAGE	O	Anzahl der Zeilen, die pro Seite gedruckt werden sollen.
FOREGROUND?(Y/N)	R	Durch Eingabe von Y(yes) wird das Kommando im Vordergrund gestartet. Bei Eingabe von N(no) wird das Kommando im Hintergrund gestartet.

2.8.7.3 Beispiel zum BPPF-Kommando

Im nachstehenden Beispiel ist angezeigt, wie ein Benutzer durch zweimaligen Aufruf des BPPF-Kommandos zwei Ausschnitte aus einer Druckdatei erstellt und sie dann auf dem Drucker LP01 ausdruckt. Der erste Ausschnitt soll auf jeder Seite in Zeile 1 Spalte 40 beginnend die Kennzeichnung COPY OF ORDERS, der zweite Ausschnitt keine Kennzeichnung erhalten. Der Benutzer in diesem Beispiel benennt seine Dateien für Druckwiederholungen nach folgendem Schema:

.S$PRINT. <User Id> nn

Dabei bezeichnet <User Id> die Benutzerkennung, die der Benutzer beim Login am Terminal angegeben (hier: GGE022) hatte und nn eine laufende Numerierung von 00 bis 99 für die erstellten Ausschnitte (hier: 04 und 05).

```
()  BPPF
    BUILD PARITIAL PRINT FILE
    ORIGINAL FILE PATHNAME: VOL01.ORDERS
    PARTIAL FILE PATHNAME: .S$PRINT.GGE02204
    FIRST PAGE: 3
    LAST PAGE: 10
    ANSI FORMAT: NO
    TEXT TO INSERT: COPY OF ORDERS
    INSERT IN LINE: 1
    COLUMN: 40
    LINES PER PAGE
    FOREGROUND? (Y/N): Y

()  BPPF
    BUILD PARTIAL PRINT FILE
    ORIGINAL FILE PATHNAME: VOL01.ORDERS
    PARTIAL FILE PATHNAME: .S$PRINT.GGE02205
    FIRST PAGE: 20
    LAST PAGE: 35
    ANSI FORMAT: NO
    TEXT TO INSERT:
    INSERT IN LINE:
    COLUMN:
    LINES PER PAGE:
    FOREGROUND? (Y/N): Y

()  PF
    PRINT FILE
    FILE PATHNAME(S): .S$PRINT.GG02204, .S$PRINT.GGE02205
    ANSI FORMAT?: NO
    LISTING DEVICE: LP01
    DELETE AFTER PRINTING?: YES
    NUMBER OF LINES/PAGE:
    FORM:
    PREFERRED PRINTING: NO
    NUMBER OF COPIES: 1
    SHOW OUTPUT STATUS: NO

()
```

2.8.8 History File des Spooler Systems

Der TAXO Spooler führt ein Protokoll über alle Druckaufträge, bei denen beim PF-Kommando ein anderes Löschkennzeichen (DELETE AFTER PRINTING) als Y oder N angegeben wurde. Dieses Protokoll wird als History File bezeichnet. Der History File hat den Pfadnamen .S$PRINT.HISTORY und kann mit dem S$PF (Show Printed Files) Kommando am Terminal angezeigt werden. Mit Hilfe des D$PF (Delete Printed Files) Kommando können zu einem beliebigen Zeitpunkt Dateien, die mit einem bestimmten Löschkennzeichen versehen sind, gelöscht werden.

2.8.8.1 Anzeigen des Spooler History Files

S$PF - SHOW PRINTED FILES

Das S$PF-Kommando zeigt den Spooler History File .S$PRINT.HISTORY an dem Bildschirm an, an dem es aufgerufen wurde. In der Anzeige kann wie beim SF (Sow File) Kommando "geblättert" werden.

2.8.8.1.1 Benutzerantworten beim S$PF-Kommando

Das Kommando erfordert keine Eingaben des Benutzers.

2.8.8.1.2 Beispiel zum S$PF-Kommando

```
( ) S$PF
    SHOW PRINTED FILES

        SYS000  LP03  3  A  .S$PRINT.SYS00022
        SYS001  LP01     C  .S$PRINT.SYS00115
        SYS000  LP01  2  B  VOL01.ORDERS

          (1)     (2) (3)(4)        (5)
```

Erläuterungen zum Beispiel:

Jede Zeile zeigt einen durch PF (PRINT FILE) erstellten Druckauftrag. Die mit (1) ... (5) bezeichneten Felder enthalten folgende Angaben:

(1) Die Benutzerkennung des Benutzers, der den Druckauftrag erzeugte.

(2) Der Name des Gerätes, auf dem gedruckt werden sollte.

(3) Die Formularnummer, die beim Druckauftrag angegeben wurde.

(4) Das bei DELETE AFTER PRINTING angegebene Löschkennzeichen.

(5) Der Name der Datei, die gedruckt werden sollte.

2.8.8.2 Löschen von im History File enthaltenen Druckdateien

D$PF - Delete Printed Files

Das D$PF (Delete Printed Files) Kommando dient zum Löschen von Druckdateien, die im History File mit einem bestimmten Löschkennzeichen versehen sind. Das D$PF-Kommando arbeitet mit einer Kopie des History Files im .S$PRINT.HISTWRK. Der Spooler History File .S$PRINT.HISTORY wird durch D$PF neu angelegt und steht für neue Einträge damit wieder zur Verfügung. Das D$PF-Kommando liest jeden Druckauftrag aus der Arbeitsdatei (.S$PRINT.HISTWRK) und vergleicht das eingetragene Löschkennzeichen mit dem vom Benutzer beim D$PF Aufruf vorgegebenen. Bei Übereinstimmung wird versucht, die Datei und den Eintrag aus dem History File zu löschen. Ist das Löschen nicht möglich oder stimmen die Löschkennzeichen nicht überein, wird der Eintrag in den neuen History File .S$PRINT. HISTORY zurückgeschrieben.

2.8.8.2.1 Benutzerantworten zum D$PF-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
DELETE MARK	R	Löschkennzeichen. Die im History File mit diesem Zeichen versehenen Druckdateien werden gelöscht.

2.8.8.2.2 Beispiel zum D$PF-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer einen Druckauftrag, der das Löschkennzeichen "C" hat sowie die zugehörige Druckdatei mit D$PF löschen kann.

```
( ) S$PF
    SHOW PRINTED FILES

        SYS000  LP03    3  A  .S$PRINT.SYS00022
        SYS001  LP01       C  .S$PRINT.SYS00115
        SYS000  LP01    2  B  VOL01.ORDERS

( ) D$PF
    DELETE PRINTED FILES
        DELETE MARK: C

( ) S$PF
    SHOW PRINTED FILES

        SYS000  LP03    3  A  .S$PRINT.SYS00022
        SYS000  LP01    2  B  VOL01.ORDERS
```

2.8.9 Einstellen der Druckbreite (LPWIDTH)

Mit diesem Kommando kann die Puffergröße (= Zeichen je Zeile) für einen Drucker eingestellt werden. Der zulässige Wertebereich liegt zwischen 1 bis 253 Zeichen.

2.8.9.1 Format des LPWIDTH-Kommandos

( ) LPWIDTH
LINE PRINTER PAPER WIDTH
LINE PRINTER NAME: <name>
PAPER WIDTH: <int>

2.8.9.2 Antworten des Benutzers beim LPWIDTH-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Antwort des Benutzers
LINE PRINTER NAME:	R	Gerätename des Druckers, für den die Puffergröße eingestellt werden soll. Standardwert: LP01
PAPER WIDTH	R	Puffergröße des gewählten Druckers. Wertebereich zwischen 1 und 253 Zeichen. Standardwert: 80

2.8.9.3 Beispiel für ein LPWIDTH-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer bei einem Drucker, der als LP02 angeschlossen ist, die Druckbreite auf 210 Zeichen/Zeile einstellen kann:

```
( ) LPWIDTH
    LINE PRINTER PAPER WIDTH
    LINE PRINTER NAME: LP02
    PAPER WIDTH: 210
```

2.8.10 International Print File Kommando (IPF)

Das IPF Kommando (International Print File) formatiert Ausgabedateien gemäß den Erfordernissen des benutzten Typenrads bzw. Druckertyps, und trägt die Datei in die Ausgabewarteschlange für den gewählten Drucker ein. Dies setzt eine Internationisierung des Druckers voraus, um den Zeichensatz des jeweiligen Landes unterstützen zu können.

Zur Verfügung stehen Konvertierungstabellen für die Vorgabewerte; diese Tabellen werden benutzt, um TAXO-Dateien für die Ausgabe auf einen Drucker, mit einem Typenrad für FRANCE oder DEUTSCHLAND, vorzubereiten. Für andere, von TAXO unterstützte Länder ist keine Konvertierung notwendig, wenn die für das jeweilige Land normierte, internationale Tastatur in Verbindung mit dem entsprechenden landesspezifischen Typenrad benutzt wird. Benutzer können Konvertierungsdateien für andere Typenräder, Drucker oder Sprachen, sowie für das Überstreichen, selbst erstellen.

Benutzer können Ausgabe-Formatierdateien für das Anpassen von Ursprungsdaten an die Code-Erfordernisse eines beliebigen Druckertyps, oder um Überstreichzeichen zu spezifizieren, selbst erstellen. Solche Formatierdateien müssen in dem folgenden Datenformat erstellt werden:

Satz	Datei Spalte 1	
1	XX	Ländercode (zwei Zeichen)
2	YY 1,1	Buffer-Koeffizient * 10 (zwei Zeichen) * Buffer Koeffizient: max. Ausgabelänge / max. Eingabelänge (<140)
3	Z1,Z2, . . .Zn	Liste der Zeichen, die von Rückwärtsschritten beeinflußt werden. Kein Eintrag bedeutet, daß kein Zeichen davon beeinflußt wird.
4	VALUE 0	Austauschcode für ASCII > 00
5	VALUE 1	Austauschcode für ASCII > 01
.		
.		
.		
131		

2.8.10.1 Format des IPF Kommandos

```
IPF
INTERNATIONAL PRINT FILE
INPUT FILE PATHNAME: <acnm>@
OUTPUT FILE PATHNAME: <acnm>@
COUNTRY: <acnm>@
ANSI FORMAT?: [ yes/no ]
LISTING DEVICE: <acnm>@
DELETE INPUT AFTER PRINTING?:[ yes/no ]
DELETE OUTPUT AFTER PRINTING?: [ yes/no ]
NUMBER OF LINES/PAGE: [ int ]
```

2.8.10.2 Benutzerantworten zum IPF Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT FILE PATHNAME:	R	Pfadname der auszudruckenden Datei.
OUTPUT FILE PATHNAME:	R	Der Pfadname einer Datei, wohin die Information ausgegeben werden soll.
COUNTRY:	R	Der Pfadname einer Formatiertabelle, die die Ausgabedatei in das entsprechende Format konvertiert. Falls die Datei mit der Formatiertabelle unter dem Directory .S$CNTRY steht und als eigentlicher File-Name nur einer oder zwei Buchstaben verwendet werden (z.B.: US Dir. .S$CNTRY.US) kann hier statt des vollständigen Pfadnamens auch der "eigentliche" Dateiname angegeben werden. Z. Zt. liegen unter der Directory .S$CNTRY zwei Files mit Namen F und D. Bei Eingabe von F wird die Formatierungstabelle für das Typenrad FRANCE benutzt. In diesem Fall werden die Zeichen Circumflex ( ) und Umlaut (" ") über jedes ihnen in der Ausgabedatei vorausgehende Zeichen gedruckt. Wird D eingegeben, so wird die Formatiertabelle für das Typenrad DEUTSCHLAND benutzt. Dieses paßt den in TAXO benutzten deutschen Zeichensatz an den Drucker an (siehe Anhang B für eine Beschreibung der Zeichensätze). Die beiden Formattabellen sind keineswegs allgemeingültig, sondern auf den beim Test vorhandenen Drucker ausgerichtet.
ANSI FORMAT?:	O	Y (yes) bedeutet, daß die Datei ANSI Steuerzeichen enthält. NO bedeutet, daß die Datei keine ANSI Steuerzeichen enthält. Das ANSI Steuerzeichen (der erste Teil eines Satzes) hat die folgende Bedeutung: Blank - 1-zeiliger Vorschub vor dem Drucken 0 - 2-zeiliger Vorschub vor dem Drucken 1 - Seitenvorschub vor dem Drucken + - Kein Zeilenvorschub (Überdrucken) Der Vorgabewert ist YES.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LISTING DEVICE:	O	Der Gerätename eines Geräts für die Druckausgabe des Dateiinhaltes. Falls kein Gerätename angegeben ist, wird nur die Ausgabedatei erstellt.
DELETE INPUT AFTER PRINTING?:	O	Y (yes) bedeutet, daß die ursprüngliche Eingabedatei nach dem Formatieren gelöscht werden soll. NO bedeutet, daß die ursprüngliche Datei nicht gelöscht werden soll. Der Vorgabewert ist NO.
DELETE OUTPUT AFTER PRINTING?:	O	Y (yes) bedeutet, daß die Datei, die ausgedruckt wird, nach Abschluß des Druckvorganges gelöscht werden soll. NO bedeutet, daß die Datei nicht gelöscht werden soll. Der Vorgabewert ist YES.
NUMBER OF LINES/PAGE?:	O	Die Anzahl Zeilen, die auf eine Seite gedruckt werden soll. Der Vorgabewert ist 62.

2.8.10.3 Beispiel zum IPF Kommando

Das folgende Beispiel zeigt, wie ein Benutzer den Inhalt der Datei VOL1.DIR2.FRENCH1 auf einem Drucker mit 55 Zeilen pro Seite ausdrucken kann.

```
[ ] IPF
    INTERNATIONAL PRINT FILE
    INPUT FILE PATHNAME: VOL1.DIR2.FRENCH1
    OUTPUT FILE PATHNAME: .TIXAMP
    COUNTRY: F
    ANSI FORMAT?: YES
    LISTING DEVICE: LP01
    DELETE INPUT AFTER PRINTING?: NO
    DELETE OUTPUT AFTER PRINTING?:YES
    NUMBER OF LINES/PAGE: 55

                Test d'un texte français
                utilisant les mots clêfs
                suivants:

                Noël, élève, tête, fran-
                çais, gîte, hôtel, à, où,
                était, pâte, mais, flûte.

                La tabutation utilisée
                est de 20 à 50.
```

A Noël les élèves en tête
de classe de français
eurent un gîte fourni
dans un hôtel à Nice, où
le menu principal était à
base de pâtes et de mais
avec heureusement une
flûte de Champagne.

Ce texte doit continuer
sur la page suivante.

2.8.10.4 Fehlermeldungen vom IPF Kommando

Die folgenden Fehlermeldungen werden nur vom IPF Kommando ausgegeben.

Fehlercode	Text	Grund
01	IPF ERROR	Der Puffer-Koeffizient ist nicht numerisch.
02	IPF ERROR	Der Puffer-Koeffizient ist kleiner als 10.
03	COUNTRY FILE ERROR	Die Konvertierungstabelle beinhaltet weder 128 noch 256 Einträge.
04	INPUT FILE ERROR	Die Eingabedatei ist nicht sequentiell.
05	IPF ERROR	Ein unbekanntes Zeichen wurde in der Eingabedatei gefunden.

2.8.11 Erzeugen eines Druckauftrags aus einem Anwendungsprogramm

TAXO V 2.0.0 bietet die Möglichkeit, den TAXO Spooler aus einer Anwendertask anzusprechen. (Print File Funktion in Anwendertask). Dies ist wie folgt möglich:

Das Anwenderprogramm muß eine LUNO Zuweisung und einen Open auf einen logischen Drucker durchführen. Hierdurch wird auf der Systemplatte im Directory .S$PRINT automatisch eine Datei angelegt (Dateiname konfliktfrei bezüglich anderer Tasks und Spoolprozesse)

Das Anwendugsprogramm schreibt mit WRITE Anweisungen für die zugewiesene LUNO Druckzeilen in diese Datei.

Bei CLOSE mit nachfolgendem Release LUNO wird die erstellte Datei in die Spooler Auftragswarteschlange für den gewünschten Drucker eingereiht. Es ist nicht möglich, diese Druckaufträge mit einer Formularnummer zu versehen und die Anzahl der Kopien zu beeinflussen (Formularnummer ist immer '0' und 'NUMBER OF COPIES' ist '1').

Die folgende Tabelle zeigt die verfügbaren logischen Druckernamen und ihre Zuordnung zu den im System vorhandenen Druckern:

Logischer Name	Tatsächlicher Drucker
LP$1	Systemdrucker LP01
LP$2	Systemdrucker LP02
"	"
"	"
"	"
LP$9	Systemdrucker LP09

Die Verwendung von logischen Druckernamen in COBOL Programmen wird im COBOL Handbuch beschrieben.

Anmerkung:

Aus technischen Gründen können auf die beschriebene Art nur Druckaufträge für die angegebenen Systemdrucker erzeugt werden.

2.8.12 Formularsteuerdateien

Für eine Formularnummer kann eine Steuerdatei spezifiziert werden. Sie enthält Code Sequenzen, die spezielle Funktionen des Druckers ein- oder ausschalten (z. B. Sperrschrift, Schmalschrift, Plotmodus, Einstellen der Formularlänge u. ä.). Es ist zu unterscheiden zwischen Code Sequenzen, die global auf einen Drucker wirken (z. B. Formularlängeneinstellung) d. h. sie werden einmalig an den Drucker ausgegeben und die gewünschte Funktion ist eingestellt, und solchen, die zeilenlokal wirken, d. h. sie müssen mit jeder Druckzeile an den Drucker ausgegeben werden.

Mit Hilfe einer Steuerdatei kann der TAXO Spooler beide Typen von Steuercode Sequenzen an einen Drucker ausgeben. Dabei werden globale Code Sequenzen einmalig beim Formularwechsel mit dem Kommando MSD (Modify Spooler Device) ausgegeben. Sind in der Formularsteuerdatei auch zeilenlokale Code Sequenzen spezifiziert, werden diese mit jeder Druckzeile der Druckaufträge ausgegeben, die für die Formularnummer vorhanden sind.

Anmerkung:

Druckaufträge ohne angegebene Formularnummer bekommen vom Spooler die Formularnummer '0' zugewiesen. Existiert dafür eine Formularsteuerdatei, wird diese auch verwendet für Druckaufträge, die ohne Formularnummer erzeugt sind.

2.8.12.1 Aufbau von Formularsteuerdateien

Formularsteuerdateien können mit dem Texteditor erstellt werden. Sie müssen auf der aktuellen Systemplatte für den Spooler abgelegt werden.

2.8.12.1.1 Pfadnamen für Formularsteuerdateien

Alle Formularsteuerdateien müssen im Directory .S$FORMS abgelegt sein. Der Dateiname besteht aus dem Buchstaben 'F' gefolgt von der Formularnummer (5 stellig, führende Nullen sind notwendig, wenn Formularnummer weniger Stellen besitzt).

Beispiele: Pfadname der Formularsteuerdatei für Formularnummer

```
12345: S$FORMS.F12345
   12: S$FORMS.F00012
    0: S$FORMS.F00000
```

2.8.12.1.2 Format von Formularsteuerdateien

Die mit dem Texteditor erstellbaren Formularsteuerdateien müssen folgendes Format haben:

Beliebige Anzahl von Zeilen

jede Zeile kann eine oder mehrere Steuercode Sequenzen vom gleichen Typ (global oder zeilenlokal) definieren

Zeilen mit globalen und zeilenlokalen Code Sequenzen können beliebig gemischt werden

es können beliebig Kommentarzeilen eingefügt sein

Spalte 1 - 2	Typ der folgenden Code Sequenze	
	#G global;	wird beim Formularwechsel einmalig an den Drucker ausgegeben.
	#L zeilenlokal;	wird bei allen Druckaufträgen für die Formularnummer am Anfang jeder Druckzeile ausgegeben.
	Enthalten die Spalten 1 - 2 andere Zeichen wird die Zeile als Kommentarzeile betrachtet und ignoriert.	
Spalte 3 - 80	Code Sequenz in Form einer Liste von Ganzzahlen (Integerzahlen). Die Listenelemente müssen durch Komma getrennt sein. Jedes Listenelement muß im Bereich	
	0 . . . > FFFFFFFF	
	liegen, d. h. durch 32 Bits darstellbar sein.	

<u>Beispiel:</u>

>7F, >01, >05	gleichbedeutend mit
>7F01, >05	gleichbedeutend mit
>7F0105	gleichbedeutend mit
>7F, >0105	gleichbedeutend mit
127, >0105	

2.8.13 Ausgabe und Behandlung von Fehlermeldungen des Spoolers

Vom Spooler werden auf die Systemzeile Statusmeldungen und Fehler beim Abarbeiten von Druckaufträgen ausgegeben. Die Meldungen erscheinen auf dem Terminal, von dem aus der Druckauftrag mit PF (PrintFile) aus dem Vorder- und Hintergrund erzeugt wurde.
Spooler Meldungen auf die Systemzeile eines Terminals haben die Form:

<Spool Id> <Text>

mit

<Spool Id>		die dem Druckauftrag durch PF zugewiesene Spool Id
<Text >	einer der folgenden Texte:	
	HALTED	Das Ausdrucken des Auftrags wurde durch HO (Halt Output) unterbrochen.
	RESUMED	Das Ausdrucken des Auftrags wurde durch RO (Resume Output) wieder fortgesetzt.
	KILLED	Der Druckauftrag wurde durch KO (Kill Output) abgebrochen bzw. aus der Warteschlange gelöscht.
	INP E CCEE	Beim Zugriff auf die auszudruckende Datei trat bei der I/O Operation CC der Fehler EE auf. Siehe Kap. 6 im Fehlerhandbuch.
	OUT E CCEE	Beim Zugriff auf den Drucker trat bei der I/O Operation CC der Fehler EE auf. Siehe Kapitel 6 im Fehlerhandbuch.
	FRM E CCEE	Bei der Verarbeitung der Formularsteuerdatei trat der Fehler EE auf. Ist CC größer oder gleich >80, so handelt es sich um einen Fehler, der bei Aufruf einer SCI Routine gemeldet wurde (Fehlerbeschreibung: Kap. 2 im Fehlerhandbuch). Ist CC kleiner >80, so trat der Fehler bei der I/O Operation CC auf. (Siehe Kap.6 im Fehlerhandbuch).

Die Meldungen des Spoolers auf der Systemzeile eines Terminals müssen vom Benutzer zwar durch Drücken von Shift + Delete quittiert werden. Der Spooler wartet jedoch nicht auf diese Quittierung, sondern arbeitet normal weiter.

Anmerkungen

1. Alle Spoolermeldungen auf die Systemzeile werden zusätzlich ins System-Logbuch geschrieben.

2. Für Druckaufträge, die über einen logischen Drucker erzeugt wurden (LP$1 ...LP$9), werden keine Meldungen an das Benutzerterminal gesendet. Sie erscheinen nur im System-Logbuch.

Im Allgemeinen bewirkt das Auftreten einer Fehlerbedingung einen Abbruch des jeweiligen Drucktauftrags. Nur in folgenden Fällen wird der Druckauftrag weiter bearbeitet:

- Drucker meldet Time Out
  Fehlermeldung: OUT E 0B06
  Der Druck wird nach Online Sezen des Druckers forgesetzt.

- Fehler beim Verarbeiter der Formularsteuerdatei
  Fehlermeldung: FRM E CCEE
  Der DRuck wird ohne die in der Formularsteuerdatei angegebenen Code Sequenzen ausgeführt.

3 KOPIEREN VON DATEIEN UND DIRECTORIES

3.1 Allgemeines

Die folgenden Kommandos erlauben den Inhalt von Dateien oder Directories zu kopieren und die erhaltenen Kopien zu prüfen:

- CKSR (Copy Kif to Sequential File Randomly)
- CPI (Copy Program Image)
- MKL (Modify Kif Logging)
- CC (Copy Concatenate)
- AF (Append File)
- CKS (Copy Key indexed file to Sequential file)
- CSK (Copy Sequential file to Key indexed file)
- Directory Utility Kommandos

  CD (Copy Directory)
  VC (Verify Copy)
  BD (Backup Directory)
  RD (Restore Directory)
  VB (Verify Backup or restore)

Dieses Kapitel beschreibt diese Kommandos und zeigt anhand von Beispielen, wie sie benutzt werden. Darüber hinaus beschreibt dieses Kapitel den Gebrauch von Kontrolldateien für Directory Utility Kommandos. Kontrolldateien sind Dateien, die Anweisungen und Optionen dafür enthalten, wie Copy Kommandos ein Directory kopieren sollen.

3.2 Kopieren und Verketten von Dateien (CC Kommando)

Mit dem Kommando CC (Copy Concatenate) kann man eine oder mehrere Dateien von physikalischen Geräten oder vom Plattenspeicher in eine Datei kopieren, sie zusammenbinden und so als eine Datei unter einem einzigen Dateinamen greifbar machen. Die einzigen Einschränkungen hinsichtlich der Menge der Dateien, welche mit einem CC Kommando zusammengebunden werden können, ist gegeben durch die Gesamtlänge der Pfadnamen oder Gerätenamen (50 Zeichen insgesamt), welche jene Dateien bzw. die Träger jener Dateien kennzeichnen, die zusammengefügt werden sollen. Durch Verwendung von Synonymen für Pfadnamen oder Gerätenamen kann hierbei Platz gespart werden.

Besteht die Ausgabedatei noch nicht, so wird sie von Copy Concatenate automatisch, gemäß der ersten Eingabedatei, angelegt. Erfolgt die Eingabe über ein Gerät, so wird die Ausgabedatei sequentiell, erweiterbar und mit Leerzeichenunterdrückung angelegt, mit der logischen Satzlänge von 80 Bytes. Erfolgt die Eingabe von einer Datei, so wird die Ausgabedatei mit den gleichen Merkmalen angelegt, ausgenommen, daß die vorhandene Gesamtzahl der Sätze der Eingabedatei die primäre Platzzuordnung für die Ausgabedatei bestimmt und die sekundäre Bereichszuordnung vorgegeben wird. Die maximale Satzlänge gibt an, wie viele Bytes beim Kopieren per I/O Operation übertragen werden. Sie wirkt sich nicht auf die Satzlänge der Ausgabedatei aus. Weder Schreibschutz, noch Löschschutz, noch zwingend vorgeschriebene Schreibcharakteristika werden mit übertragen.

Das CC Kommando kann auch benutzt werden, ebenso wie das Kommando PF (Print File), um eine Datei auf einem Zeilendrucker oder irgend ein anderes Ausgabegerät auszugeben. Benötigt ein Ausgabegerät Transportkontrollen, so fügt das CC Kommando einzelne Zwischenraumkontrollen für die Ausgabe hinzu, bis die zu kopierende Datei Transportkontrollzeichen aufweist. Dann werden diese Zeichen benutzt. Dieses Kommando kann sowohl im Vordergrund als auch im Hintergrund abgearbeitet werden.

ANMERKUNG:

Copy Concatenate ist nicht in der Lage, Directories, Programmdateien, Image Dateien oder Key Index Dateien zu übertragen. Um solche Dateien zu übertragen, muß das Kommando CD (Copy Directory) benutzt werden.

3.2.1 Format des CC Kommandos

```
CC
COPY/CONCATENATE
INPUT ACCESS NAME(S): (<acnm>) @
OUTPUT ACCESS NAME: <acnm> @
REPLACE?: <yes/no>
MAXIMUM RECORD LENGTH: [ int ]
FOREGROUND? (Y/N): <yes/no> @
```

3.2.2 Benutzerantworten beim CC Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT ACCESS NAME(S):	R	ein oder mehrere Pfadnamen oder Gerätenamen (getrennt durch Kommata), die eine oder mehrere Dateien spizifizieren, welche in einer Ausgabedatei zusammengefügt werden sollen. Die Reihenfolge der Angaben der Pfadnamen und Gerätenamen bestimmt, in welcher Folge die entsprechenden Dateien zusammengefügt werden. Die Anzahl der Dateien, die aneinandergekettet werden können, wird durch die Länge des Antwortfeldes (50 Zeichen) begrenzt, in welches Pfadnamen, Gerätenamen bzw. deren Synonyme eingegeben werden müssen.
OUTPUT ACCESS NAME:	R	Der Gerätename, bzw. der Pfadname der Datei, auf welche TAXO die zusammengefügten Dateien ausgeben soll. Falls der Pfadname keine schon vorhandene Datei benennt, legt Copy Concatenate diese automatisch an. Der angegebene Pfadname oder Gerätename muß sich vom Input Access Name unterscheiden.
REPLACE?:	R	Angabe Y (yes) bewirkt, daß eine eventuell schon vorhandene Ausgabedatei des gleichen Namens von der zu übertragenden Datei überschrieben würde. Angabe N (no) verhindert die Ausführung des Kommandos, falls die angegebene Ausgabedatei schon existiert. Keine Antwort bedeutet N.
MAXIMUM RECORD LENGTH:	O	die maximale Länge der zu kopierenden Sätze. Standardwert ist 1024 Bytes.
FOREGROUND? (Y/N):	R	Die Eingabe von Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N(NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

3.3 Anhängen einer oder mehrerer Dateien an eine andere Datei (AF Kommando)

Mit dem Kommando AF (Append File) kann man einer Datei eine andere anfügen oder weitere Eingaben über ein sequentielles Eingabegerät einer Datei hinzufügen. Sequentielle Plattendateien, relative Plattendateien oder Eingaben von sequentiellen Geräten können an eine sequentielle Plattendatei, eine relative Plattendatei oder einer Ausgabe auf ein Magnetband angehängt werden. Eingaben, die an eine Datei gehängt werden, werden unmittelbar nach dem letzten Satz der Datei angefügt, wobei die letzte EOF-Marke in der Datei überschrieben (zerstört) wird. Den angehängten Sätzen wird eine neue EOF-Marke hinzugefügt. Directories, Programmdateien, Image Dateien und Key Index Dateien lassen sich durch Append File nicht anhängen. Dieses Kommando kann sowohl im Vordergrund als auch im Hintergrund abgearbeitet werden.

3.3.1 Format des AF Kommandos

```
AF
APPEND FILE
INPUT ACCESS NAME(S): (<acnm>) @
OUTPUT PATHNAME: <acnm> @
MAXIMUM RECORD LENGTH: [ int ]
FOREGROUND (Y/N): <yes/no> @
```

3.3.2 Benutzerantworten beim AF Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT ACCESS NAME(S):	R	Ein oder mehrere Pfadnamen oder Gerätenamen (durch Kommata getrennt), welche die Quelle der Information angeben, die an die Ausgabedatei angehängt oder auf ein Magnetband kopiert werden soll. Die Anzahl der Dateien, die angehängt werden können, wird begrenzt durch die Menge der Pfadnamen, Gerätenamen oder Synonyme, welche in dem 50 Zeichen langen Antwortfeld untergebracht werden können.
OUTPUT PATHNAME:	R	Ein Pfadname der Datei, an welche die angegebene Datei bzw. Dateien oder die Informationseingabe von einem sequentiellen Gerät angehängt werden soll. Existiert die angegebene Ausgabedatei noch nicht, wird sie automatisch angelegt.
MAXIMUM RECORD LENGTH:	O	Die maximale Länge der zu kopierenden Sätze. Standardwert ist 1024 Bytes.
FOREGROUND (Y/N):	R	Die Eingabe von Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N(NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

3.3.3 Beispiel zum AF Kommando

Das folgende Beispiel zeigt den Inhalt von vier Dateien, das Anhängen dreier von ihnen an die erste, und den Ausdruck des neuen Inhalts der ersten Datei.

```
() SF

SHOW FILE
 FILE PATHNAME:     .TEST

RECORD NUMBER ONE
RECORD NUMBER TWO
RECORD NUMBER THREE
()

() SF

SHOW FILE
 FILE PATHNAME: VOL1.FOUR

RECORD NUMBER FOUR
()

() SF

SHOW FILE
 FILE PATHNAME: VOL1.FIVE

RECORD NUMBER FIVE
()

() SF

SHOW FILE
 FILE PATHNAME: VOL1.SIX

RECORD NUMBER SIX
()

() AF

APPEND FILE
 INPUT ACCESS NAME(S):    VOL1.FOUR,VOL1.FIVE,VOL1.SIX

 OUTPUT PATHNAME:    .TEST

 MAXIMUM RECORD LENGTH:    80
 FOREGROUND (Y/N): YES
APPEND COMPLETED
()

() SF

SHOW FILE
 FILE PATHNAME: .TEST

RECORD NUMBER ONE
RECORD NUMBER TWO
RECORD NUMBER THREE
RECORD NUMBER FOUR
RECORD NUMBER FIVE
RECORD NUMBER SIX
()
```

3.4 Modifizieren der Aufzeichnung bei Key Index Dateien (MKL Kommando)

Das MKL Kommando dient dem einleitenden Laden eines Satzes bei Key Indexed Dateien. Hierzu wird das modifizierte Bit "Schreiben" des KIF FDR (File Descriptor Record) ausgeschaltet. Dies geschieht dadurch, daß die bestehende Datei in eine zeitlich verlagerte (asynchrone) Schreibdatei geändert wird.

Das MKL Kommando sollte nach dem Erstellen der Key Indexed Dateien angewandt werden, jedoch nur, falls ein einleitendes Satzladen mittels des CSK Kommandos erfolgt ist.
MKL unterdrückt die sogenannte Vorab-Aufzeichnung der Sätze und das Aktualisieren der freien Speicherhinweisadresse des FDR zu dem Zeitpunkt, wenn die Datei eröffnet wird.
Eine Close Anweisung schaltet die Vorab-Aufzeichnung wieder ein. (Erfolgt direkt vom System)

Jede Verarbeitung, die den Inhalt der Datei verändert, (Einfügen, Zurückschreiben, Löschen) wird bedeutend schneller durchgeführt, wenn sich die Datei im zeitlich verlagerten (asynchronen) Schreibmodus befindet.
Hierbei sind auf die Platte weniger Lese- und Schreibzyklen notwendig.

Die nachstehende Tabelle zeigt die evtl. erforderlichen Plattenzugriffe während der einleitenden Verarbeitung einer einzelnen Key Indexed Datei mit Baumstruktur lediglich mit Root Node und Leaf Node. (Erklärung siehe (1) und (2)).
Schlüssel werden in sequentieller Folge eingefügt.

Tabelle: Erforderliche Plattenzugriffe für die einleitende Erstellung von Key Indexed Dateien

Ausführung	Schreiben modifiziert	Schreiben asynchron
Lesen Datensatz	M	M
Vorab Aufzeichnen Datensatz	Y	N
Schreiben Datensatz	Y	M
Lesen (1) Root Node	M	M
Lesen (2) Leaf Node	M	M
Vorab Aufzeichnen Leaf Node	Y	N
Schreiben Leaf Node	Y	M
Verändern FDR	Y	M

1. Knoten dienen zum Auffinden der einzelnen logischen Sätze der Datei
2. Das Element am Ende eines Zweiges heißt Blatt. (Leaf Node)

Die Einträge in den Spalten haben folgende Bedeutung:

M: Der Satz braucht nicht von der Platte gelesen zu werden, da er sich im Speicher befindet. Der Satz wird nur auf die Platte geschrieben, falls die Datei geschlossen ist bzw. andere Buffer oder Tasks den Speicher belegen.
(Im Falle FDR wird der Satz dann auf die Platte geschrieben, wenn die der Datei zugewiesene Luno freigegeben ist.)

N: Eine Verarbeitung ist nicht durchgeführt worden.

Y: Der Satz muß von der Platte gelesen oder auf die Platte geschrieben werden.

WARNUNG:

Die beschriebene Key Indexed Datei ist unbrauchbar, falls das MKL Kommando wegen eines Systemabsturzes oder aus einem anderen ungewöhnlichen Grund nicht vollständig ausgeführt werden kann. Die Datei ist zu löschen. Danach muß sie neu erstellt werden. MKL Kommando nochmals eingeben.

3.4.1 Format des MKL Kommandos

MKL
MODIFY KIF LOGGING
KEY INDEX FILE:<acnm>@

3.4.2 Benutzerantworten zum MKL Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
KEY INDEX FILE:	R	Der Pfadname der für das Urladen vorzubereitenden Key Indexed Datei.

3.4.3 Beispiel zum MKL Kommando

Das folgende Beispiel zeigt das Erstellen einer Key Indexed Datei und das Vorbereiten der Datei auf Urladen der Sätze mittels MKL.

```
[] CFKEY
CREATE KEY INDEXED FILE
 PATHNAME:    VOL1.KIF
 LOGICAL RECORD LENGTH:     80
 PHYSICAL RECORD LENGTH:     >240
 INITIAL ALLOCATION:    100
 SECONDARY ALLOCATION:    200
 MAXIMUM SIZE:     125
KEY DESCRIPTION FOR KEY NUMBER 1
 START POSITION:    1
 KEY LENGTH:    5
 DUPLICATES?: NO
 MODIFIABLE?: NO
 ANY MORE KEYS?:    YES
KEY DESCRIPTION FOR KEY NUMBER 2
 START POSITION:    1    8
 KEY LENGTH:    4
 DUPLICATES?: NO    Y
 MODIFIABLE?: NO
 ANY MORE KEYS?:   YES . NO
[]
[] MKL
MODIFY KIF LOGGING
 KEY INDEX FILE:     VOL1.KIF
[]
```

3.5 Übertragen sequentieller Dateien in Key Index Dateien (CSK Kommando)

Mit dem Kommando CSK (Copy Sequential to Key Indexed File) wird der Inhalt einer sequentiellen Datei in eine zuvor definierte Key Index Datei übertragen. Es ist möglich, dieses Kommando sowohl im Vordergrund als auch im Hintergrund abzuarbeiten.

3.5.1 Format des CSK Kommandos

CSK
COPY SEQUENTIAL FILE TO KIF
INPUT ACCESS NAME: <acnm> @
OUTPUT PATHNAME: <acnm> @
FOREGROUND (Y/N): <yes/no>

3.5.2 Benutzerantworten beim CSK Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT ACCESS NAME:	R	der Gerätenamen oder der Pfadnamme der sequentiellen Datei deren Informationsinhalt sequentiell auf die noch anzugebende Key Index Datei übertragen werden soll.
OUTPUT PATHNAME:	R	Ein Pfadname der Key Index Datei, die die sequentiellen Informationen aufnehmen soll. Die Key Index Datei muß vor Eingabe des CSK Kommandos angelegt werden!
FOREGROUND (Y/N):	R	Durch die Eingabe von Y(YES) wird das Kommando im Vordergrund abgearbeitet. Durch die Eingabe von N(NO) wird das Kommando im Hintergrund abgearbeitet.

3.5.3 Beispiel zum CSK Kommando

Das folgende Beispiel zeigt das Anlegen einer Key Index Datei, den Ausdruck des Inhalts einer sequentiellen Datei und das Übertragen einer sequentiellen Datei in eine Key Index Datei mittels eines CSK Kommandos.

```
() CFKEY
CREATE KEY INDEXED FILE
 PATHNAME:    VOL1.KIF
 LOGICAL RECORD LENGTH:    80
 PHYSICAL RECORD LENGTH:    >240
 INITIAL ALLOCATION:    100
 SECONDARY ALLOCATION:    200
 MAXIMUM SIZE:    125
KEY DESCRIPTION -
 START POSITION: 0  1
 KEY LENGTH:    5
 DUPLICATES?: NO
 MODIFIABLE?: NO
 ANY MORE KEYS?: YES
KEY DESCRIPTION
 START POSITION: 1  8
 KEY LENGTH:    4
 DUPLICATES?: NO  Y
 MODIFIABLE?: NO
 ANY MORE KEYS?: YES
KEY DESCRIPTION
 START POSITION: 8  14
 KEY LENGTH:    2
```

```
 DUPLICATES?: NO  Y
 MODIFIABLE?: NO  Y
 ANY MORE KEYS?: YES  NO
( )
( ) CSK
COPY SEQUENTIAL FILE TO KIF
 INPUT ACCESS NAME: VOL1.KIFSRC
 OUTPUT PATHNAME: VOL1.KIF
 FOREGROUND (Y/N): YES
( )
( ) SF
SHOW FILE
 FILE PATHNAME: VOL1.KIFSRC
05243  0027  15  REGULAR WIDGET
05289  0153  07  SUPER WIDGET
05314  0092  04  MICRO WIDGET
( )
```

3.6 Übertragung von Key Index Dateien nach sequentiellen Dateien (CKS Kommando)

Mit dem Kommando CKS (Copy Key Indexed file to Sequential file) kann man den Inhalt einer Key Index Datei in eine sequentielle Datei übertragen, wobei der Benutzer die Art der Übertragung entscheidet. Er muß dafür den Schlüssel angeben, der die Reihenfolge der zu lesenden Sätze während der Übertragung in die sequentielle Datei angibt. Gibt der Benutzer beispielsweise auf die Anfrage nach dem Key die Nummer 2 ein und dieser Schlüssel enthält den PLZ Code so wird die Key Index Datei sequentiell nach dem PLZ Code gelesen und so in die sequentielle Datei übertragen. Ist das Gerät STXX oder LPXX, werden nur die ersten 80 Zeichen eines Satzes übertragen. Dieses Kommando kann sowohl im Vordergrund als auch im Hintergrund abgearbeitet werden.

ANMERKUNG:

Das Kommando CKS (Copy Key Indexed File to Sequential File) sollte nicht benutzt werden, um Key Index Dateien zu sichern. Um eine Key Index Datei zu sichern, benutze man das Kommando Copy Directory (CD) (siehe Paragraph 3.6.1) oder Backup Directory (BD), wobei als Input der Pfadname der zu sichernden Datei anzugeben ist.

3.6.1 Format des CKS Kommandos

```
CKS
COPY KIF TO SEQUENTIAL FILE
INPUT PATHNAME: <acnm> @
KEY: <integer>
BEGINNING KEY VALUE: <String >
ENDING KEY VALUE:    <String >
ASCII OR BINARY (A/B): <A/B >
OUTPUT ACCESS NAME: <acnm> @
REPLACE?: <yes/no>
FOREGROUND (Y/N): <yes/no>
```

3.6.2 Benutzerantworten zum CKS Kommando

Systemmeldungen	Anwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT PATHNAME:	R	Der Pfadname der zu übertragenden Key Index Datei.
KEY:	R	Eine Ganzzahl n ($1 \leq n \leq 14$), welche die Nummer des Schlüssels in der Key Index Datei angibt, der die Reihenfolge der Sätze bei der Übertragung in eine sequentielle Datei angibt.
BEGINNING KEY-VALUE	O	Die Key Datei wird ab einem Schlüsselwert der größer oder gleich dem angegebenen Schlüsselwert ist, kopiert. Die Anzahl der Zeichen muß nicht dieselbe wie die des definierten Schlüssels sein. Falls weniger Zeichen angegeben werden, so wird der Rest des Schlüsselwertes mit binären Nullen aufgefüllt. Erfolgt keine Antwort, wird bei dem niedrigsten Schlüsselwert mit dem Kopieren begonnen.
ENDING KEY VALUE:	O	Das logische Kopieren einer Key Indexed Datei wird bei einem Schlüsselwert größer als angegeben, beendet. Die Anzahl der Zeichen kann vom definierten Schlüssel abweichen. Falls weniger Zeichen angegeben werden, wird der Rest des Schlüsselwertes mit binären Nullen aufgefüllt. Erfolgt keine Antwort, wird das Kopieren gestoppt, wenn alle logischen Sätze kopiert sind.
ASCII OR BINARY (A/B):	R	Der betreffende Schlüssel wird entweder als ASCII Zeichen oder als binärer Wert interpretiert. A : ASCII B : BINARY Binärwerte können nur die Zeichen 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, u. F enthalten. Falls der gewünschte Schlüssel sowohl ASCII als auch binäre Zeichen enthält, muß "B" eingegeben und irgendwelche ASCII Zeichen als binär spezifiziert werden. Z.B. Falls das erste Byte eines Schlüssels eine binäre Zahl >2 C und das zweite Byte der Buchstabe C ist, muß der Schlüsselwert als 2 C 43 angegeben werden.

OUTPUT ACCESS NAME:	R	Der Gerätename oder Pfadname der Datei, auf welche die Key Index Datei übertragen werden soll. Das Betriebssystem legt selbst eine sequentielle Datei an, falls die angegebene Datei vor Eingabe des CKS Kommandos noch nicht existiert. Ist das Gerät STXX oder LPXX, werden nur die ersten 80 Zeichen eines Satzes auf das Gerät übertragen.
REPLACE?:	R	Y (yes) bewirkt, daß der Inhalt der Zieldatei durch den Inhalt der zu übertragenden Datei ersetzt wird. Eingabe von N (no) verhindert die Ausführung des Kommandos im Falle, die Zieldatei existiert schon.
FOREGROUND (Y/N):	R	Durch die Eingabe von Y(YES) wird das Kommando im Vordergrund abgearbeitet. Durch die Eingabe von N(NO) wird das Kommando im Hintergrund abgearbeitet.

3.6.3 Beispiel zum CKS Kommando

Das schon früher in diesem Kapitel gezeigte Beispiel zum CKS Kommando zeigt die Übertragung einer sequentiellen Datei in eine Key Index Datei. Das nun folgende Beispiel zeigt die Übertragung einer Key Index Datei in eine sequentielle Datei. Da im gezeigten CKS Kommando die Übertragung durch den zweiten Key bestimmt wird, unterscheidet sich die Reihenfolge in der sequentiellen Datei nun von der ursprünglichen, wie sie im Beispiel zum CSK Kommando gezeigt wurde.

```
( ) CKS
    COPY KIF TO SEQUENTIAL FILE
    INPUT PATHNAME: VOL1.KIF
    KEY: 2
    BEGINNING KEY VALUE:
    ENDING KEY VALUE:
    ASCII OR BINARY (A/B): A
    OUTPUT ACCESS NAME: VOL1.KIFKEY2
    REPLACE?: NO
    FOREGROUND (Y/N): YES
    3 RECORDS OUTPUT TO FILE
( )

( ) SF
    SHOW FILE
    FILE PATHNAME: VOL1.KIFKEY2
    05243 0027  15 REGULAR WIDGET
    05314 0092  04 MICRO WIDGET
    05289 0153 07 SUPER WIDGET
( )
```

3.7 Copying Key Indexed Files to Sequential Files Randomly (das CKSR Kommando)

Das CKSR Kommando (Copy Key Indexed File to Sequential Randomly) kopiert den Inhalt einer Key-Indexed-Datei in eine sequentielle Datei oder auf ein sequentielles Gerät. Dateien werden nicht in einer bestimmten Reihenfolge kopiert. Der Zweck dieses Dienstprogrammes ist es, möglichst viele Daten von einer teilweise zerstörten bzw. von einer wegen E/A-Fehlern unbrauchbar gewordenen Key-Indexed-Datei zu retten. Das Dienstprogramm liest die physikalischen Sätze der Key-Indexed-Datei, und überprüft die Gültigkeit der Sätze im Datenblock. Die logischen Sätze werden der Datei entnommen und in eine sequentielle Datei geschrieben. Ein Satz, der keinen gültigen Datenblock darstellt, führt zu einem Widerspruch beim Entnehmen eines logischen Satzes bzw. zu einem Fehler beim Lesen des Datenblocks. Wurde ein Widerspruch festgestellt, wird die physische Satznummer des fehlerhaften Blockes, der Offset im fehlerhaften Block und eine Beschreibung des Fehlers in die Listdatei geschrieben.

ANMERKUNG:

Im Falle einer einfachen Key-Indexed-Datei kann das Dienstprogramm nur für die Bestimmung von fehlerhaften Blöcken benutzt werden. Die extrahierten Sätze enthalten keine Schlüssel, da in einer solchen Datei der Schlüssel nicht mit im Satz gespeichert wird. Die fehlerhaften Blöcke müssen "manuell" korrigiert und CKSR neu aufgerufen werden.

Dieses Kommando kann nur für Key Indexed Dateien mit Organisationsform "Sequential Placement" verwendet werden.

3.7.1 Format des CKSR Kommandos

```
CKSR
COPY KIF TO SEQUENTIAL RANDOMLY
KEY INDEXED FILE PATHNAME: <acnm>@
SEQUENTIAL FILE PATHNAME: <acnm>@
LISTING ACCESS NAME: <acnm>@
FOREGROUND? (Y/N): <yes/no >
```

3.7.2 Benutzerantworten zum CKSR Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
KEY INDEXED FILE PATHNAME:	R	Der Pfadname der zu kopierenden Key Indexed Datei.
SEQUENTIAL FILE PATHNAME:	R	Der Pfadname der sequentiellen Datei in welche die logischen Sätze geschrieben werden sollen. Die Datei wird automatisch angelegt wenn sie nicht bereits existiert.
LISTING ACCESS NAME:	O	Der Pfadname der Datei oder des Gerätes, wohin eine Liste der fehlerhaften Blöcke der Key Indexed Datei geschrieben werden soll. Bei Eingabe von Null wird die Liste in die TLF des Terminals geschrieben.
FOREGROUND? (Y/N):	R	Durch die Eingabe von Y(Yes) wird das Kommando im Vordergrund abgearbeitet. Bei Eingabe von N(No) erfolgt die Kommandoverarbeitung im Hintergrund.

3.7.3 Beispiel zum CKSR Kommando

Das folgende Beispiel zeigt die Ausführung des CKSR Kommandos für eine Key Indexed Datei, mit fünf physischen Sätzen. Wenn das CKSR Kommando oder ein Benutzerprogramm versuchen würde, auf sie zuzugreifen, könnte dies zu Fehlern oder zu einem Systemsturz führen.

```
( ) CKSR
COPY KIF TO SEQUENTIAL RANDOMLY
  KEY INDEXED FILE PATHNAME: VOL1.KIF
  SEQUENTIAL FILE PATHNAME: VOL1.SEQ
  LISTING ACCESS NAME: VOL1.LIST
  FOREGROUND? (Y/N): YES

1809          LOGICAL RECORDS COPIED
( )

( ) SF
SHOW FILE
  FILE PATHNAME: VOL1.LIST
RECORD   OFFSET          MESSAGE
 00062    00D2    LOGICAL RECORD SIZE TOO LARGE
 00063    00F0    KEY ID TOO LARGE
 00064    0000    INCORRECT RECORD NUMBER
 00080    00F2    UNSUPPRESSION ERROR
 0022B    00EE    LOGICAL RECORD SIZE IS ODD

1809          LOGICAL RECORDS COPIED
( )
```

3.7.4 CKSR Listdatei-Meldungen

Tabelle 6.2 enthält die Listdatei-Meldungen für das CKSR Kommando.

Tabelle 6.2 CKSR Listdatei-Meldungen

Meldung	Bedeutung
INCORRECT RECORD NUMBER	Die ersten zwei Worte des physischen Satzes sind mit der physischen Satznummer nicht identisch.
LOGICAL RECORD SIZE IS ODD	Die mit Leerstellenunterdrückung aufgezeichnete logische Satzlänge, muß eine gerade Zahl sein.
LOGICAL RECORD SIZE TOO LARGE	Die logische Satzlänge gilt als zu groß, wenn sie die definierte Satzlänge um mehr als 50 Prozent überschreitet.
KEY ID TOO LARGE	Die Schlüssel-ID darf die maximale Schlüssel-ID des physischen Satzes nicht überschreiten.
UNSUPPRESSION ERROR	Der Satz ist während des Rückgängigmachens der Leerstellenunterdrückung zu groß geworden.
OOXX READ ERROR	Der angegebene Fehlercode (XX) wurde beim Lesen des physischen Satzes gemeldet.
XXXXXX LOGICAL RECORDS COPIED	Die angegebene Anzahl (XXXXXX) logischer Sätze wurde von der Key Indexed Datei kopiert. Die Darstellung erfolgt im Dezimalzahlformat.

3.8 Copy Program Image (CPI Kommando)

Das CPI (Copy Program Image) Kommando kopiert das Abbild einer Task, einer Prozedur oder eines Overlays von einer Programmdatei auf eine andere. Das Kommando kann das Abbild auch auf einen anderen Platz innerhalb der gleichen Programmdatei übertragen. Wenn das Abbild einer Task kopiert wird, es ist möglich, sämtliche mit der Task verbundene Prozeduren und/oder Overlays ebenfalls zu kopieren, es sei denn, das Abbild wird auf einen anderen Platz innerhalb der Programmdatei verlegt. Der Name und/oder die ID des Abbildes in der Ausgabedatei darf von dem Namen oder der ID in der Eingabedatei abweichen. Der Name und die ID müssen jedoch eindeutig sein, wenn das Abbild auf einen anderen Platz innerhalb der gleichen Programmdatei kopiert wird. Die Namen und die IDs der zusammen mit der Task zu übertragenden Prozeduren und Overlays müssen identisch sein. Sollte es wünschenswert sein, die Namen und/oder die IDs zu ändern, so müssen die Prozeduren und Overlays getrennt kopiert werden.

Wurden die zu einer Task gehörigen Overlays zusammen mit der Task kopiert, dann wird ein Auszug (MPF, Map Program File) der Ausgabedatei die Overlays in der umgekehrten Reihenfolge der Eingabe zeigen. Dies sollte für den Benutzer jedoch kein Problem darstellen.

3.8.1 Format des CPI Kommandos

```
CPI
COPY PROGRAM IMAGE
INPUT PROGRAM FILE: <acnm>@
OUTPUT PROGRAM FILE: [ acnm] @
INPUT IMAGE NAME OF ID: <string>
OUTPUT IMAGE NAME AND/OR ID: <string>
IMAGE TYPE (T/P/O): <T/P/O>
OPTIONS: <ADD/REPLACE>
```

Die folgenden zwei Meldungen werden angezeigt, wenn für IMAGE TYPE ein T eingegeben und ein Pfadname für die Ausgabe-Programmdatei spezifiziert wird:

```
COPY PROGRAM IMAGE TASK EXTENSION
COPY ASSOCIATED OVERLAYS?: <Y/N>
COPY ASSOCIATED PROCEDURES?: <Y/N>
```

Die folgende Meldung wird angezeigt, wenn für IMAGE TYPE ein O eingegeben wird:

```
COPY PROGRAM IMAGE OVERLAY EXTENSION
ASSOCIATED TASK NAME OR ID: <string>
```

3.8.2 Benutzerantworten zum CPI Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT PROGRAM FILE:	R	Der Pfadname der Programmdatei, die das zu kopierende Abbild enthält.
OUTPUT PROGRAM FILE:	O	Der Pfadname der Programmdatei, in die das Abbild kopiert werden soll. Keine Eingabe wenn das Abbild in die gleiche Programmdatei einkopiert werden soll.

INPUT IMAGE NAME OR ID:	R	Der Name oder die ID des zu kopierenden Abbildes.
OUTPUT IMAGE NAME AND/OR ID:	O	Der Name und/oder die ID unter welche(n) das Abbild in die Ausgabe-Programmdatei installiert werden soll. Wird keine Eingabe gemacht, so wird der Name und die ID von der Eingabedatei übernommen; wird nur die ID eingegeben, so wird ebenfalls der Name von der Eingabedatei übernommen. Werden beide eingegeben, so müssen sie durch ein Komma voneinander getrennt sein. Wird für OUTPUT PROGRAM FILE keine Eingabe gemacht, so müssen der Name und die ID für die Eingabedatei eindeutig sein.
IMAGE TYPE (T/P/O):	R	T eingeben, wenn es sich um das Abbild einer Task handelt; P eingeben, wenn es sich um das Abbild einer Prozedur handelt; O eingeben, wenn es sich um das Abbild eines Overlay handelt.
OPTIONS:	R	REPLACE eingeben, wenn das Abbild in der Ausgabedatei bereits vorhanden ist und ersetzte werden soll; andernfalls ADD eingeben.
COPY ASSOCIATED OVERLAYS?:	R	Y eingeben, wenn sämtliche mit der Task verbundenen Overlays ebenfalls kopiert werden sollen; andernfalls N eingeben.
COPY ASSOCIATED PROCEDURES?:	R	Y eingeben, wenn alle mit der Task verbundenen Prozeduren ebenfalls kopiert werden sollen; andernfalls N eingeben.
ASSOCIATED TASK NAME OR ID:	R	Eingabe des Namens oder TASK-ID in der Ausgabe-Programmdatei, in welche das Overlay einkopiert werden soll.

3.8.3 Beispiele zum CPI Kommando

Das erste Beispiel kopiert einen Task, zusammen mit ihren Overlays und Prozeduren, von einer Programmdatei in eine andere, und installiert die Task auf der Ausgabe-Programmdatei gemäß ihrer ID. Das zweite Beispiel kopierte das angegebene Overlay von einer Programmdatei in eine andere, installiert das Overlay in die Ausgabe-Programmdatei gemäß seinem Namen und seiner ID, und verbindet das Overlay mit einer bestimmten Task. Das dritte Beispiel kopiert die angegebene Prozedur von einer Programmdatei in eine andere, und installiert die Prozedur auf die Ausgabe-Programmdatei unter Benutzung der in INPUT IMAGE NAME OR ID: angegebenen ID.

```
( ) CPI
COPY PROGRAM IMAGE
 INPUT PROGRAM FILE:          .S$PROGA
 OUTPUT PROGRAM FILE:         .EXAMPLE
 INPUT IMAGE NAME OR ID:          >33
 OUTPUT IMAGE NAME AND/OR ID:        >01
 IMAGE TYPE (T/P/O):            T
 OPTIONS: ADD
COPY PROGRAM IMAGE TASK EXTENSION
 COPY ASSOCIATED OVERLAYS?: Y
 COPY ASSOCIATED PROCEDURES?: Y

( ) CPI
COPY PROGRAM IMAGE
 INPUT IMAGE NAME OR ID:  .S$PROGA
 OUTPUT PROGRAM FILE:         .EXAMPLE
 INPUT IMAGE NAME OR ID:          >3
 OUTPUT IMAGE NAME AND/OR ID:        >7,NEW
 IMAGE TYPE (T/P/O):            O
 OPTIONS: ADD
COPY PROGRAM IMAGE OVERLAY EXTENSION
 ASSOCIATED TASK NAME OR ID:      >1

( ) CPI
COPY PROGRM IMAGE
 INPUT PROGRAM FILE:          .S$PROGA
 OUTPUT PROGRAM FILE:         .EXAMPLE
 INPUT IMAGE NAME OR ID:                >2
 OUTPUT IMAGE NAME AND/OR ID:
 IMAGE TYPE (T/P/O):            P
 OPTIONS: ADD
```

3.9 Directory Kommandos

Die folgenden Directory Utility Kommandos sind zum Kopieren, Sichern, Zurückschreiben und Prüfen von Directories oder 'set of files' vorgesehen:

- CD (Copy Directory) Kommando
- VC (Verify Copy) Kommando
- BD (Backup Directory) Kommando
- RD (Restore Directory) Kommando
- VB (Verify Backup or Restore) Kommando

ANMERKUNG:

Für CD, BD, RD, VC und VB dürfen Pfadnamen nicht mit Plattennamen beginnen, wie z.B. DS02.S$PROC. Benutzen Sie bitte Plattennamen wie MYDISC.S$PROC, wenn die Dateien nicht auf der Systemplatte liegen. Nur direkte, ungeblockte Ein/Ausgabe-operationen werden unterstützt.

Ein 'Set of Files' enthält alle Dateien und Directories (falls vorhanden) eines gegebenen Directories. Ein 'Set of Files' muß wenigstens ein Directory und darf höchstens alle Dateien und Directories einer Platte enthalten. Abbildung 3-1 zum Beispiel zeigt ein Directory .A, dessen 'Set of Files' aus allen Dateien und Directories besteht, die auf einer Platte namens VOL1 gespeichert sind. Die Dateien und Directories in diesem Dateienset sind durch die Pfadnamen bestimmt, wie im Bild 3-1 gezeigt.

Eine Directory Copy Operation (Copy Directory, Backup Directory oder Restore Directory) kann den Dateienset .A nach dem Dateienset .B übertragen (wobei Dateienset .B vor der Übertragung noch nicht vorhanden sein muß).

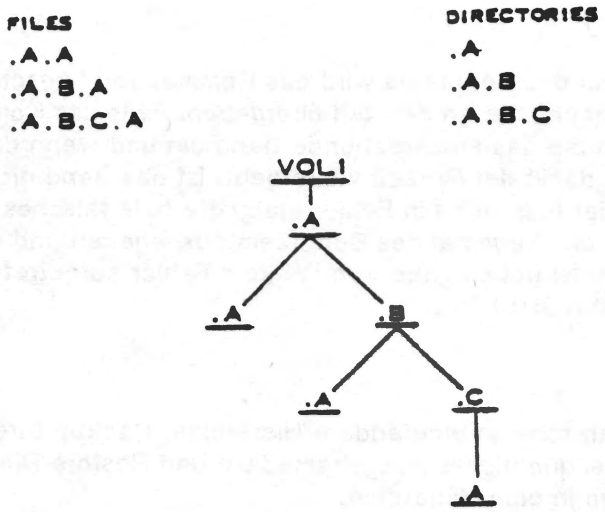


Bild 3-1 Beispiel eines Dateienset eines Directories

Diese Copy Anweisung generiert das folgende Dateienset für Directory .B wie in Bild 3-2 gezeigt.

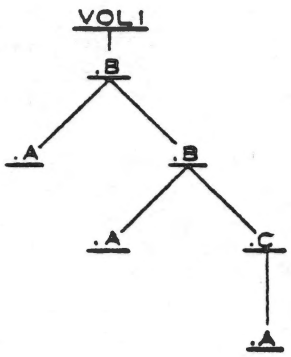


Bild 3-2 Beispiel für ein Dateienset eines Directories nach Ausführung eines CD Kommandos

Das Ergebnis der Copy Operation kann überprüft werden mittels eines Verify Kommandos, welches den Dateienset des einen Directories mit dem Dateienset eines anderen Directories vergleicht und die Gemeinsamkeiten feststellt.

Backup Directory, Restore Directory und Verify Backup sind Kommandos, die Directories, welche sich über mehr als ein Magnetband erstrecken, bearbeiten können. Dies läßt sich durch die Option MULTI erreichen, die, wenn angegeben, den ersten Satz eines jeden Bandes mit einem Header Satz für Identifikationszwecke versorgt. Der Header Satz besteht aus acht Worten wie folgt:

- Die ASCII Zeichen **HDR*
- Zeit und Datum des Backup Directories (in vier Worten)
- Die Bandnummer

Der Header Satz wird bei Angabe der NOMULTI OPTION nicht geschrieben und verursacht einen Fehler, falls sich das Directory über mehr als ein Magnetband erstreckt. Ist die MULTI OPTION gegeben und das Ende des Bandes erreicht, erscheint auf dem Bildschirm des Benutzers folgende Meldung:

"MOUNT TAPE X . TYPE $ TO QUIT, Y TO CONTINUE"

wobei X die Bandnummer bedeutet.

Durch Drücken der $-Taste am Terminal des Benutzers wird das Kommando abgeschlossen und die Durchführkontrolle bis zum Prozeßende an den SCI übergeben. Falls das Kommando nicht abgeschlossen werden soll, legen Sie das entsprechende Band ein und wenn das Band richtig positioniert ist, drücken Sie 'Y', damit der Prozeß weitergeht. Ist das Band nicht richtig positioniert (wenn Sie 'Y' eingeben) oder falls sich ein Fehler ereignete (wie falsches Band eingelegt), wird die Fehlermeldung auf das Terminal des Benutzers ausgegeben und die Meldung, das Band einzulegen, neu ausgegeben. Ist bei Eingabe von 'Y' kein Fehler aufgetreten, wird der Prozeß fortgeführt. Vorgabewert ist NOMULTI.

ANMERKUNG:

Copy Directory kopiert eine Hierarchie in eine andere Hierarchie, Backup Directory kopiert eine Hierarchie auf ein sequentielles Ausgabemedium und Restore Directory kopiert ein sequentielles Medium in eine Hierarchie.

3.9.1 Kopieren von Directories (CD Kommando)

Das CD (Copy Directory) Kommando erlaubt dem Benutzer ein 'Set of Files' eines Directory in ein anderes Directory zu übertragen und zu verifizieren. Es ist möglich, dieses Kommando im Vordergrund als auch im Hintergrund abzuarbeiten. Der Inhalt des Quelldirectory wird aufgrund dieser Copy Operation nicht verändert. Dateien und Directories, die im Zieldirectory nicht vorhanden sind, werden durch das CD Kommando automatisch angelegt.

Das CD Kommando kopiert alle Dateien und alternativen Namen eines Directory (mit Ausnahme der Dateien .S$ROLLA, .S$CRASH, .S$PRINT, und .S$DIAG, wenn nicht das CD Kommando vom Benutzer durch Anweisungen und Optionen eingeschränkt wird. Sollen die vom Kopiervorgang ausgeschlossenen Dateien .S$ROLLA, .S$CRASH, .S$PRINT und .S$DIAG trotzdem mit auf die Zielplatte übernommen werden, so müssen sie entweder durch eine INCLUDE-Anweisung im Control File oder durch die Angabe bei INPUT PATHNAME in einem eigenen Programmlauf kopiert werden. Sie werden ebenfalls kopiert, wenn sie sich in einem Subdirectory befinden. Anweisungen ergänzen das CD Kommando in Form einer Kontrolldatei, wie nachfolgend beschrieben. Kontrolldateien können die folgenden Directiven einschließen: MOVE, EXCLUDE, INCLUDE und OPTION. Options sind entweder verbunden mit einer Option Anweisung, die in einer Kontrolldatei oder in der Antwort auf eine Systemmeldung nach Eingabe des CD Kommandos enthalten sein kann. ADD, ALIASES, NOALIASES oder REPLACE sind Optionen, die in beiden Fällen auftreten können. Das CD Kommando kann das Ergebnis seiner Ausführungen auf ein vom Benutzer zu bestimmendes Gerät oder in einer Datei protokollieren.

Über den Aufrufparameter KIF COMPRESS? (Y/N) hat der Benutzer Einfluß auf das Kopieren von Key Indexed Dateien die nach dem "Sequential PLacement" Verfahren organisiert sind. Diese können je nach Parametereingabe auf dem Zielmedium entweder verkleinert (entsprechend dem aktuellen Datenbestand) oder in ursprünglicher Größe angelegt werden. Key Indexed Dateien, die nach dem "Hashed Placement" Verfahren organisiert sind, beeinflußt dieser Parameter nicht.

Der Benutzer kann über einen Aufrufparameter steuern, ob die erstellte Kopie mit dem Original verglichen (verifiziert) werden soll. Dies geschieht durch einen automatischen Aufruf des Verify Copy Kommandos mit den bei CD eingegebenen Parametern.

Beim Verifizieren wird ein Dateienset eines Master Directories mit einem Dateienset einer Directory Kopie verglichen und festgestellt, welche Dateien in beiden gleich sind. Es werden Filetyp, Gebrauch der Datei, Dateinamen und Dateiinhalt auf entsprechendem Level eines jeden Sets verglichen und Gemeinsamkeiten aufgedeckt. Das Ergebnis dieses Vergleiches wird auf ein Ausgabegerät oder in einer Datei, die vom Benutzer angegeben wird, abgelegt.

Wurde eine Kontrolldatei angegeben, schränkt dies die Dateien im Master Directory, welches mit der Directory Kopie verglichen wurde, in der gleichen Weise ein, wie oben bereits beschrieben.

ANMERKUNGEN:

Obwohl Kontrolldatei, Eingabe Pfadnamen sowie Ausgabe Pfadnamen auf die entsprechenden Systemanfragen nicht angegeben werden müssen, ist es nötig, zumindest die Kontrolldatei anzugeben, wenn die beiden anderen nicht genannt werden. Beide, Eingabe- und Ausgabe Pfadnamen, müssen eingegeben werden, wenn die Kontrolldatei fehlt.

3.9.1.1 Format des CD Kommandos

CD
COPY DIRECTORY
INPUT PATHNAME: [acnm] @
OUTPUT PATHNAME: [acnm] @
CONTROL ACCESS NAME: [acnm] @
LISTING ACCESS NAME: [acnm] @
OPTIONS: [(string)]
KIF COMPRESS (Y/N): <yes/no >
FOREGROUND (Y/N): <yes/no>
VERIFY (Y/N): <yes/no>

VERIFY ACCESS NAME: [acnm] @ — wird nur dann angezeigt, wenn bei der vorangehenden Meldung mit "YES" geantwortet wurde.

3.9.1.2 Benutzerantworten zum CD Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT PATHNAME	O	Ein Pfadname des Directory, das das ganze Dateienset beinhaltet, welches kopiert und gegebenenfalls auch verifiziert werden soll. Es kann auch der Pfadname einer einzelnen Datei sein, wenn nur diese Datei kopiert (und verifiziert) werden soll. Wird kein Input Pfadname eingegeben, muß ein Control Access Name benannt werden.
OUTPUT PATHNAME	O	Der Pfadname eines Directory, auf das die Datei bzw. der 'Set of Files', wie in INPUT PATHNAME angegeben, übertragen und gegebenenfalls verglichen werden soll. Falls beim Kopieren das für die Ausgabe angegebene Directory nicht existiert, wird das Directory höchster Stufe analog der Größe des Directory angelegt, das kopiert werden soll. Wird kein OUTPUT PFADNAME angegeben, muß ein CONTROL ACCESS NAME angegeben werden.

Vorsicht:

Der OUTPUT PATHNAME muß ein Directory bezeichnen. Z. B. um eine Datei .C.A in eine Datei .B.A zu kopieren, geben Sie .C.A als den Input Pfadnamen und .B als Output Pfadnamen an. Nehmen Sie nicht .B.A als OUTPUT PATHNAME. Dies impliziert das Directory .B.A. Der resultierende Pfadname der Datei würde dann lauten .B.A.A.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
CONTROL ACCESS NAME	O	Der Gerätename oder Pfadname der Datei, in welcher die Kontrollanweisungen zur Überwachung der Operationen stehen. Diese Anweisungen, die in einem Control Stream eingebunden sein können, werden im Kap. 3.10 beschrieben. Wird kein Control Access Name angegeben, müssen sowohl Input Pfadname als auch Output Pfadname benannt werden.

Vorsicht:

Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei CONTROL ACCESS NAME 'ME' unzulässig.

Im Vordergrund ist die Eingabe von 'ME' erlaubt. Sollte auch noch verifiziert werden, so sind die Kontrollanweisungen für den Verifizierungslauf nach dem Kopieren zu wiederholen. Der Kopiervorgang ist beendet, wenn der Cursor erneut am Bildschirm erscheint.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LISTING ACCESS NAME	O	Der Gerätename oder Pfadname einer Datei, auf welche(n) eine Zusammenfassung der Ergebnisse der Copy Operationen aufgelistet werden soll. Wird im Vordergrund keine Antwort angegeben, so gilt der Terminal Local File des benutzten Terminals.

Vorsicht:

Soll das Kommando im Hintergrund abgearbeitet werden, so darf das Kontroll-Listing weder auf dem Bildschirm noch im Terminal Local File protokolliert werden.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
OPTIONS	O	Eine oder mehrere der folgenden Optionen können - getrennt durch Kommata - eingegeben werden. Damit kann festgelegt werden, ob alternative Dateinamen (Alias Namen) mitkopiert werden sollen oder nicht, ob Dateien, die im Zieldirectory bereits vorhanden sind, ersetzt werden soll, ob Dateien erst ab einem bestimmten letzten Updatedatum kopiert werden sollen und ob sequentielle oder relative Dateien eine neue physikalische Satzlänge erhalten sollen. Mögliche Optionen: ADD ALIASES NOALIASES REPLACE DATE SPRL RPRL NSPRL NORPRL Vorgabewerte: ADD, ALIASES Diese Optionen werden im Paragraph 3.10 näher beschrieben.

KIF COMPRESS? (Y/N)	O	Die Eingabe von Y (YES) bewirkt, daß eine oder mehrere Key Indexed Dateien, die unter dem bei INPUT PATHNAME eingegebenen Directory liegen, bzw. dort selbst angegeben sind, auf dem Zielmedium (OUTPUT PATHNAME) nur noch so groß angelegt werden, wie dies zur Aufnahme des aktuellen Datenbestandes erforderlich ist. Bei Eingabe von N (NO) jedoch bleibt die ursprüngliche Größe der Key Indexed Datei(en) erhalten.

Anmerkung:

Dieser Parameter hat jedoch nur Einfluß auf das Kopieren von Key Indexed Dateien, die nach dem "Sequential Placement" Verfahren organisiert sind. Key Indexed Dateien, die nach dem "Hashed Placement" Verfahren organisiert sind, werden davon nicht berührt.

FOREGROUND (Y/N)	R	Y (YES) zeigt an, daß der Kopiervorgang und gegebenenfalls auch der Verifizierungsvorgang im Vordergrund ablaufen soll. N (NO) zeigt an, daß der Kopiervorgang und gegebenenfalls auch der Verifizierungsvorgang im Hintergrund ablaufen soll.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
VERIFY (Y/N)	R	Die Eingabe von Y (YES) bewirkt, daß nach dem Kopiervorgang sofort auch verifiziert wird. Durch die Eingabe von N (NO) wird nur kopiert, nicht verifiziert.
VERIFY ACCESS NAME	O	Der Name eines Gerätes oder der Pfadname einer Datei, auf welche die Ergebnisse des Vergleiches ausgegeben werden sollen. Vorgabewert ist der Terminal Local File des Benutzerterminals.

Vorsicht:

Soll der Verifizierungslauf im Hintergrund laufen, so darf das Kontroll-Listing weder auf dem Bildschirm noch im Terminal Local File protokolliert werden.

Es ist darauf zu achten, daß die Angabe bei VERIFY ACCESS NAME mit der bei LISTING ACCESS NAME nicht identisch ist, sofern es sich um Pfadnamen von Dateien handelt, sonst wird das Kontroll-Listing des Kopiervorganges überschrieben.

Falls bei der Systemmeldung OPTIONS als Benutzerantwort DATE eingegeben wurde, so erscheinen neben der Meldung "CD: DATE OF OLDEST FILE" noch folgende Systemmeldungen:

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
YEAR	R	Eingabe des Jahres in dezimaler oder hexadezimaler Form.
MONTH	R	Eingabe des Monats in dezimaler oder hexadezimaler Form.
DAY	R	Eingabe des Tages in dezimaler oder hexadezimaler Form.
HOUR	O	Eingabe der Stunde in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als 0 Stunde angenommen.
MINUTE	O	Eingabe der Minute(n) in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Minute(n) 0 angenommen.

3.9.1.3 Beispiel zum CD Kommando:

Das folgende Beispiel zeigt die Eingabe eines CD Kommandos zur Sicherung des Directories CMD48. DEMO in das Directory DATA1. SAVE.

```
( ) CD
    COPY DIRECTORY
    INPUT PATHNAME: CMD48. DEMO
    OUTPUT PATHNAME: DATA1. SAVE
    CONTROL ACCESS NAME:
    LISTING ACCESS NAME: .COPLIST
    OPTIONS: ADD, ALIAS
    KIF COMPRESS? (Y/N): NO
    VERIFY (Y/N): YES
    FOREGROUND (Y/N): YES

    VERIFY ACCESS NAME: .VERLIST
```

```
COPY DIRECTORY                    14:55:03 TUESDAY, DEC 15, 1981.

ORIGINAL SOURCE:          CMD48.DEMO
ORIGINAL DESTINATION:     DATA1.SAVE
ORIGINAL OPTIONS:         ALIASES, ADD
CONTROL FILE:
LIST FILE:                 COPLIST


**        DIRECTORY  DATA1.SAVE.SYSTEM

**        DIRECTORY  DATA1.SAVE.SYSTEM.TABLES
**   PSB
**   FCB
**   TSB
**   OVT
**        END OF DIRECTORY  DATA1.SAVE.SYSTEM.TABLES

**        END OF DIRECTORY  DATA1.SAVE.SYSTEM

**   XAF$CONT  IS AN ALIAS OF  XAF
**   XAF$LIST  IS AN ALIAS OF  XAF
**   XAF       - PROGRAM FILE

**   TASK        ID
**   XAF        >01

**   PROCEDURE  ID

**   OVERLAY    ID
**        END OF PROGRAM FILE

**        DIRECTORY  DATA1.SAVE.LIST
**   TACDSR
**   QWIKCOPY
**        END OF DIRECTORY  DATA1.SAVE.LIST


**        DIRECTORY  DATA1.SAVE.SOURCE
**   TACPDT
**   TASK$A
**   TACDSR
**   QWIKCOPY
**        END OF DIRECTORY  DATA1.SAVE.SOURCE

**   S$DIAG
****** COPY DIRECTORY COMPLETED
```

```
VERIFY COPY                    14:56:11 TUESDAY, DEC 15, 1981.

**        DIRECTORY CMD48.DEMO.SYSTEM

**        DIRECTORY CMD48.DEMO.SYSTEM.TABLES
**   PSB
**   FCB
**   TSB
**   OVT
**        END OF DIRECTORY CMD48.DEMO.SYSTEM.TABLES

**        END OF DIRECTORY CMD48.DEMO.SYSTEM

**   XAF$CONT IS AN ALIAS OF XAF
**   XAF$LIST IS AN ALIAS OF XAF
**   XAF        - PROGRAM FILE

**   TASK         ID
**   XAF        >01

**   PROCEDURE  ID

**   OVERLAY    ID
**        END OF PROGRAM FILE

**        DIRECTORY CMD48.DEMO.LIST
**   TACDSR
**   QWIKCOPY
**        END OF DIRECTORY CMD48.DEMO.LIST


**        DIRECTORY CMD48.DEMO.SOURCE
**   TACPDT
**   TASK$A
**   TACDSR
**   QWIKCOPY
**        END OF DIRECTORY CMD48.DEMO.SOURCE

**   S$DIAG
****** VERIFY COPY COMPLETED
```

3.9.2 Verifizieren einer Kopie eines Directories (VC Kommando)

Das VC (Verify Copy) Kommando vergleicht ein Dateienset eines Master Directories mit einem Dateienset einer Directory Kopie und stellt fest, welche Dateien in beiden gleich sind. Das VC Kommando vergleicht Filetyp, Gebrauch der Datei, Dateinamen und Dateiinhalt von Dateien auf entsprechendem Level eines jeden Sets und deckt Gemeinsamkeiten auf. Das Ergebnis dieses Vergleiches wird auf ein Ausgabegerät ausgegeben oder in einer Datei, die vom Benutzer angegeben wird, abgelegt.

Wird eine Kontrolldatei mit einem VC Kommando angegeben, schränkt dies die Dateien im Master Directory, die mit der Directory Kopie verglichen werden, in der gleichen Weise ein, wie eine Copy Operation eines CD (Copy Directory) Kommando. Alternativnamen werden verglichen, es sei denn, NOALIASES wäre angegeben.

Dieses Kommando kann sowohl im Vordergrund als auch im Hintergrund abgearbeitet werden.

ANMERKUNG:

Obwohl die Angabe einer Kontrolldatei, eines Master Pfadnamens als auch eines Copy Pfadnamens nicht vorgeschrieben ist, muß, falls eine dieser Angaben fehlt, eine Kontrolldatei angegeben werden. Wird keine Kontrolldatei spezifiziert, muß sowohl ein Master Pfadname als auch ein Copy Pfadname angegeben werden.

3.9.2.1 Format des Kommandos VC

```
VC
VERIFY COPY
MASTER PATHNAME: [ acnm ] @
COPY PATHNAME: [ acnm ] @
CONTROL ACCESS NAME: [ acnm ] @
LISTING ACCESS NAME: [ acnm ] @
OPTIONS : [ (string) ]
FOREGROUND (Y/N): <yes/no>
```

3.9.2.2 Benutzerantworten zum VC Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
MASTER PATHNAME:	O	Ein Pfadname eines Directories, welches den Knoten des Master Set of Files bildet, mit welchen der im Copy Pfadnamen genannte Set of Files verglichen wird. Benutzer, die keinen Master Pfadnamen angeben, müssen den Control Access Name benennen.
COPY PATHNAME:	O	Ein Pfadname eines Directories, das den Knoten eines Set of Files bildet, welches mit dem Set of Files, genannt im Master Pfadnamen, verglichen wird. Wird kein Copy Pfadname angegeben, muß der Control Access Name benannt werden.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
CONTROL ACCESS NAME:	O	Der Name eines Gerätes oder der Pfadname einer Datei, welche(s) für das VC Kommando die Anweisungen enthält, die die Dateien, die mit dem Copy Directory verglichen werden sollen, im Master Directory begrenzen. Anweisungen in der Kontrolldatei haben Vorrang vor Antworten auf Systemmeldungen. Benutzer, welche keinen Control Access Name angeben, müssen sowohl Master als auch Copy Pfadnamen angeben.
LISTING ACCESS NAME:	O	Der Name eines Gerätes oder der Pfadname einer Datei, wohin das Ergebnis des Vergleiches geschrieben werden soll. Vorgabewert ist der Terminal Local File des Benutzerterminals.
FOREGROUND (Y/N):	R	Die Eingabe von Y (YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N (NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.
OPTIONS	O	Die Option DATE wird im Paragraph 3.10 beschrieben.- Der Verifiziervorgang kann mit den gleichen Optionen wie CD gesteuert werden.

Falls bei der Systemmeldung OPTIONS als Benutzerantwort DATE eingegeben wurde, so erscheinen neben der Meldung "CD: DATE OF OLDEST FILE" noch folgende Systemmeldungen:

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
YEAR	R	Eingabe des Jahres in dezimaler oder hexadezimaler Form.
MONTH	R	Eingabe des Monats in dezimaler oder hexadezimaler Form.
DAY	R	Eingabe des Tages in dezimaler oder hexadezimaler Form.
HOUR	O	Eingabe der Stunde in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Stunde 0 angenommen.
MINUTE	O	Eingabe der Minute(n) in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Minute(n) 0 angenommen.

3.9.2.3 Beispiel zum VC Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das VC Kommando anwenden kann, um eine Copy Operation, die schon früher in diesem Kapitel als Beispiel zu einem CD (Copy Directory) Kommando beschrieben wurde, zu verifizieren.

```
( ) VC
   VERIFY COPY
   MASTER PATHNAME:
   COPY PATHNAME: VOL1.SAVEMY
   CONTROL ACCESS NAME: VOL1.CONTROL
   LISTING ACCESS NAME: LP01
   OPTIONS:
   FOREGROUND (Y/N): YES
 ( )

  VERIFY COPY                    13:24:35 WEDNESDAY, AUG 10, 1977.

OPTION REPL
MOVE VOL1.MYFILES
EX LIST,CONTROL
  ** X

  **       DIRECTORY VOL1.MYFILES.OBJECT
  **  PROGA
  **       END OF DIRECTORY VOL1.MYFILES.OBJECT


  **       DIRECTORY VOL1.MYFILES.SOURCE
  **  PROGA
  **       END OF DIRECTORY VOL1.MYFILES.SOURCE

  **  PROG      - PROGRAM FILE

  **  TASK        ID
  **  PROGA      >01
  **  PROGB      >02
  **       END OF PROGRAM FILE
  **  C         IS AN EMPTY FILE
0 ADD
MOVE VOL1.HISFILES
IN     A
IN     B
IN     C
INCL   X
I      PROG
  **  X          DOES NOT VERIFY
  **  A
  **  PROG      - PROGRAM FILE

  **  TASK        ID
  **  PROGA      >01
  **       END OF PROGRAM FILE
  **  C         FILES FLAGS DIFFER
  **  C          DOES NOT VERIFY
  **  B         WAS NOT IN SOURCE'  *
END
****** VERIFY COPY COMPLETED
```

3.9.3 Anfertigen von Sicherungskopien von Directories (BD Kommando)

Das BD (Backup Directory) Kommando erlaubt ein 'Set of Files' eines Directories auf eine sequentielle Datei, eine Image Datei (Datei kann sich über mehrere Platten erstrecken) oder ein bzw. mehrere Magnetbänder in einem Format zu kopieren, das später ein Zurückkopieren der gesicherten Daten durch ein RD (Restore Directory) Kommando erlaubt.

Der Benutzer kann den Sicherungsvorgang mittels des Kommandos VB (Verify Backup) kontrollieren. Wird im BD Vorgang bei dem Parameter CONTROL ACCESS NAME ein Plattengerät oder eine Datei angegeben, in welcher Anweisungen zu dem Kopierlauf stehen, sind diese Angaben auch für das VB Kommando erforderlich, damit der Kontrollvorgang auch ordnungsgemäß durchgeführt werden kann.

Das BD Kommando kopiert alle Dateien eines Directories (außer Dateien mit den Namen .S$PRINT, .S$DIAG, .S$ROLLA, .S$CRASH, .S$LOADER, .S$OVLYA und .S$PROGA) sowie alle Alternativnamen dieser Dateien, es sei denn, der Benutzer schränkt durch Anweisungen und Optionen das BD Kommando ein. Sollen sie auf die Zielplatte übernommen werden, so müssen sie entweder durch eine INCLUDE-Anweisung im Control File oder durch die Angabe bei INPUT PATHNAME in einem eigenen Programmlauf kopiert werden. Anweisungen werden in Form einer Kontrolldatei zur Verfügung gestellt, die noch später in Paragraph 3.10 beschrieben wird. Der Operator gibt den Pfadnamen der Kontrolldatei auf eine Meldung des Systems beim BD Kommando hin ein.

Über den Aufrufparameter KIF COMPRESS? (Y/N) hat der Benutzer Einfluß auf das Kopieren von Key Indexed Dateien die nach dem "Sequential Placement" Verfahren organisiert sind. Diese können je nach Parametereingabe auf dem Zielmedium entweder verkleinert (entsprechend dem aktuellen Datenbestand) oder in ursprünglicher Größe angelegt werden. Key Indexed Dateien, die nach dem "Hashed Placement" Verfahren organisiert sind, beeinflußt dieser Parameter nicht.

Das BD Kommando dient zur Sicherung eines Directory auf eine sequentielle Datei. Es schreibt am Ende des Sicherungslaufes ein zweifaches End of File. Dieses doppelte End of File ermöglicht, daß mit Kopien des gesicherten Mediums leicht gearbeitet werden kann, unter Benutzung des CSM (Copy Sequential Medium) Kommandos und der Angabe der "Terminate on 2 EOF" Option. Deshalb ist es nötig, bei den Kommandos VB (Verify Backup) oder RD (Restore Directory) bei einem Magnetband mit mehrfachen Sicherungen, auf dem Medium zwischen den VB oder RD Kommandos Platz für einen Satz frei zu lassen, um das zweite EOF unterzubringen. Dies kann mittels des Forward Space LUNO (FL) Kommandos erreicht werden.

Kontrolldateien können die MOVE, EXCLUDE, INCLUDE und OPTION Anweisungen enthalten. Der Benutzer kann Anweisungen

a) durch eine OPTION Anweisung in einer Kontrolldatei
b) durch Beantworten einer OPTION: Systemanfrage im BD Kommando geben.

Optionen, die in jedem der Fälle angeführt werden können, sind ADD, ALIASES, NOALIASES, REPLACE, MULTI oder NOMULTI. Diese und weitere Optionen werden in Paragraph 3.10 beschrieben.

Das BD Kommando kann Sicherungen auf mehrere Platten oder Magnetbänder durchführen. Soll die Sicherung auf mehreren Platten erfolgen, so muß der Parameter SEQUENTIAL ACCESS NAME mit der Eingabe eines Plattennamen (z.B. DS02) beantwortet werden. Erfolgte diese Angabe, so erscheint auf dem Bildschirm die Meldung:

```
PATHNAME OF BACKUP FILE
                PATHNAME: <acnm>
```

Bei diesem Parameter muß die Antwort aus einem Platten- und einem Dateinamen bestehen. Dieser Platten-/Dateiname muß in einem Sicherungsvorgang bei allen Backup-Platten gleich sein.

Wird eine Liste von Platten- oder Magnetbandgeräten zur Ausgabe eingegeben, so werden die Geräte in der Reihenfolge angesprochen, in der sie eingegeben wurden. Ist ein Datenträger beschrieben, so erfolgt die Meldung:

MOUNT VOLUME X; TYPE $ TO QUIT, Y TO CONTINUE

Zur Fortsetzung des Kopiervorgangs ist "Y" einzugeben und zur Beendigung "$".
"X" gibt die Nummer des Dateiträgers an.

ANMERKUNG:

Wird bei der Sicherung ein Systemplattenlaufwerk verwendet, so muß darauf geachtet werden, daß vor Beendigung des BD-Laufes die Systemplatte wieder eingelegt wird.

Man kann eine mit dem BD Kommando erstellte Platte erst dann als Systemplatte verwenden, wenn man dafür gesorgt hat, daß alle bei diesem Kopiervorgang ausgeschlossenen Directories/Dateien (.S$ROLLA, .S$PROGA, .S$OVLYA, .S$LOADER, .S$CRASH) wieder auf der Platte vorhanden sind.

Die zur Sicherung verwendeten Platten müssen initialisiert und dürfen nicht installiert sein.

Backup Directory, Restore Directory und Verify Directory können auch bei solchen Directories angewandt werden, die sich über mehrere Magnetbänder erstrecken. Dies wird durch eine Option MULTI in diesem Kommando möglich, wobei, falls sie angegeben wurde, der erste Satz eines jeden Bandes einen Header zu Identifikationszwecken enthält. Der Header besteht aus acht Worten: den ASCII Zeichen '**HDR*', Zeit und Datum des Backup Directory (in vier Worten) sowie der Bandnummer. Falls NOMULTI angegeben ist, wird dieser Header nicht auf das Band geschrieben und es tritt ein Fehler auf, falls sich das Directory über mehr als ein Band erstreckt.

Ist die MULTI Option angegeben, erscheint, sobald das Bandende erreicht ist, am Bildschirm folgende Meldung:

"MOUNT TAPE X; TYPE $ TO QUIT, Y TO CONTINUE"

wobei "X" die Nummer des nächsten Bandes angibt. Das Kommando kann durch Drücken von "$" beendet werden. Die Kontrolle wird an den SCI zurückgegeben, ohne den Ablauf zu beenden. Im anderen Fall muß ein neues Band eingelegt und wenn dies geschehen ist, "Y" eingegeben werden. Ist das Band noch nicht in Ordnung oder trat ein anderer Fehler auf (wie z. B. eine falsche Bandnummer), wird der Fehler am Terminal angezeigt und die Meldung wiederholt. (Der Benutzer hat siebenmal die Möglichkeit, das (richtige) Band einzulegen - bis "$" eingegeben wird oder das Band richtig positioniert ist).

Wurde "Y" eingegeben und ist das Band richtig positioniert, geht der Ablauf des Kommandos weiter. Vorgabewert ist MULTI.

Wenn die Blockoption angegeben wird, wird der Benutzer aufgefordert, den Blockungsfaktor in Bytes anzugeben. Ausgabesätze werden in der angegebenen Blockgröße zusammengestellt (Standardwert: 9600). Jeder Block wird als physikalischer Satz geschrieben. Dadurch wird das Band besser ausgenutzt, da die Anzahl der Leerstellen zwischen den Blöcken reduziert wird. Die Blockoption darf nicht für Plattendateien angegeben werden, da das File Management mit den Blöcken der angegebenen Größe nicht arbeiten kann. Als Standardwert wird die NOBLOCK Option angegeben.

Das BD Kommando kann das Ergebnis auf ein Gerät auflisten oder in eine vom Benutzer angegebene Datei kopieren. Durch einen Aufrufparameter (VERIFY (Y/N)) hat der Benutzer die Möglichkeit, nach Ende des BD Kommandos automatisch das VB (Verify Backup) Kommando auszuführen, um das erstellte Backup mit dem Original zu vergleichen.

3.9.3.1 Format des Kommandos BD

BD
BACKUP DIRECTORY
DIRECTORY PATHNAME: [acnm] @
SEQUENTIAL ACCESS NAME: [acnm] @
CONTROL ACCESS NAME: [acnm] @
LISTING ACCESS NAME: [acnm] @
OPTIONS: [(string)]
KIF COMPRESS? (Y/N): [<yes/no>]
VERIFY (Y/N): <yes/no >
FOREGROUND (Y/N): <yes/no>
VERIFY ACCESS NAME: [acnm] @ — wird nur dann angezeigt, wenn bei der vorausgehenden Meldung mit "YES" geantwortet wurde.

3.9.3.2 Benutzerantworten zum Kommando BD

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
DIRECTORY PATHNAME:	O	Der Pfadname des höchsten Directories, das den 'Set of Files' umfaßt, der gesichert werden soll. Dieser Pfadname kann auch nur eine Datei bezeichnen, wenn nur eine einzige Datei zu sichern und zu verifizieren ist. Benutzer, die keinen Directory Pfadnamen angeben, müssen einen Control Access Name anführen.
SEQUENTIAL ACCESS NAME:	O	Der Name eines Gerätes oder der Pfadname einer sequentiellen Datei, auf welcher das Directory gesichert werden soll. Benutzer, die keinen Sequential Access Name angeben, müssen einen Control Access Name anführen. Gibt der Benutzer DSxx ein, erscheint am Bildschirm folgende Systemmeldung: PATHNAME OF BACKUP FILE PATHNAME: [acnm]
CONTROL ACCESS NAME:	O	Der Name des Gerätes oder der Pfadname der Datei, von welcher die Anweisungen zur Kontrolle der Sicherungsprozedur kommen (Anweisungen, die in einen Control Stream eingefügt werden können, werden später in diesem Kapitel beschrieben). Benutzer, die keinen Control Access Name eingeben, müssen sowohl einen Directory Pfadnamen als auch einen Sequential Access Name angeben.

Vorsicht

Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei CONTROL ACCESS NAME 'ME' unzulässig. Im Vordergrund ist die Eingabe von 'Me' erlaubt.

LISTING ACCESS NAME:	O	Der Name des Gerätes oder der Pfadname einer Datei, auf das eine Auflistung des Resultates der Sicherungsprozedur ausgegeben werden soll. Keine Angabe bestimmt den Terminal Local File des Benutzerbildschirms. Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei LISTING ACCESS NAME 'ME' unzulässig.
OPTIONS:	O	Angabe ALIASES bedeutet, daß alle mit den gesicherten Files verbundenen Alternativnamen auch mit gesichert werden sollen. Eingabe von NOALIASES bedeutet, daß keine alternativen Namen gesichert werden sollen. Eingabe von MULTI bedeutet, daß sich das Directory über mehrere Magnetbänder erstrecken kann. Eingabe von NOMULTI bedeutet, daß nur ein Magnetband benutzt wird. Keine Eingaben zu machen bedeutet ALIASES und MULTI. Angabe von DATE erlaubt den Kopierprozess auf solche Dateien zu beschränken, deren letztes Updatedatum größer oder gleich dem ausgegebenen Datum ist. Die Optionen BLOCK, NO BLOCK, NODATE, REWIND, NOREWIND, UNLOAD und NOUNLOAD werden im Paragraph 3.10 beschrieben. Anmerkung: Da 'NOMULTI' zukünftig nicht mehr unterstützt wird, kommt es durch diese Eingabe zu der Meldung "NOMULTI NO MORE SUPPORTED". Durch die Bestätigung mit der Eingabe von "C" (CONTINUE) wird die Abarbeitung fortgesetzt, ansonsten wird sie abgebrochen.

KIF COMPRESS? (Y/N) O

Die Eingabe von Y (YES) bewirkt, daß eine oder mehrere Key Indexed Dateien, die unter dem bei INPUT PATHNAME eingegebenen Directory liegen, bzw. dort selbst angegeben sind, auf dem Zielmedium (OUTPUT PATHNAME) nur noch so groß angelegt werden, wie dies zur Aufnahme des aktuellen Datenbestandes erforderlich ist. Bei Eingabe von N (NO) jedoch bleibt die ursprüngliche Größe der Key Indexed Datei(en) erhalten.

Anmerkung:

Dieser Parameter hat jedoch nur Einfluß auf das Kopieren von Key Indexed Dateien, die nach dem "Sequential Placement" Verfahren organisiert sind. Key Indexed Dateien, die nach dem "Hashed Placement" Verfahren organisiert sind, werden davon nicht berührt.

ANMERKUNG

Obwohl die Angaben über Kontrolldatei, Pfadname des Directories und Sequential Access Name bei den Systemmeldungen alle wahlfrei sind, muß die Kontrolldatei benannt werden, wenn eine der anderen Antworten fehlt. Sowohl Directory Pfadnamen als auch Sequential Access Namen sind zu benennen, wenn keine Kontrolldatei angegeben wurde.

VERIFY? (Y/N)	R	Die Eingabe von Y (YES) bewirkt, daß nach dem Sicherungsvorgang sofort verifiziert wird. Durch die Eingabe von N (NO) wird nur gesichert, nicht verifiziert.
FOREGROUND? (Y/N)	R	Y(YES) zeigt an, daß der Sicherungsvorgang und gegebenenfalls auch der Verifizierungsvorgang im Vordergrund ablaufen soll. Vorsicht: Werden für den Sicherungsvorgang Magnetbänder verwendet, ist hier die Eingabe von 'NO' unzulässig. N (NO) zeigt an, daß der Sicherungsvorgang und gegebenenfalls auch der Verifizierungsvorgang im Hintergrund ablaufen soll.
VERIFY ACCESS NAME	O	Der Name eines Gerätes oder der Name einer Datei, auf welche die Ergebnisse des Vergleichs ausgegeben werden sollen. Vorgabewert ist der Terminal Local File des Benutzerterminals. Dieser Parameter erscheint nur, wenn VERIFY? (Y/N) mit 'YES' beantwortet wurde. Vorsicht: Soll der Verifizierungslauf im Hintergrund laufen, so darf das Kontrolllisting weder auf dem Bildschirm noch im Terminal Local File erscheinen. Es ist darauf zu achten, daß die Angabe bei VERIFY ACCESS NAME mit der bei LISTING ACCESS NAME nicht identisch ist, sofern es sich um Pfadnamen von Dateien handelt, sonst wird das Kontrollisting des Sicherungsvorgangs überschrieben.

ANMERKUNG:

Falls bei der Systemmeldung OPTIONS als Benutzerantwort DATE eingegeben wurde, so erscheinen neben der Meldung "CD: DATE OF OLDEST FILE" noch folgende Systemmeldungen:

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
YEAR	R	Eingabe des Jahres in dezimaler oder hexadezimaler Form.
MONTH	R	Eingabe des Monats in dezimaler oder hexadezimaler Form.
DAY	R	Eingabe des Tages in dezimaler oder hexadezimaler Form.
HOUR	O	Eingabe der Stunde in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Stunde 0 angenommen.
MINUTE	O	Eingabe der Minute(n) in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Minute(n) 0 angenommen.

3.9.3.3 Beispiel zum BD Kommando

Das folgende Beispiel zeigt die Sicherung des Directories DATA1.ZENT in die sequentielle Datei CMD48.DIRX8 im Vordergrund mit Verifizierlauf. Für LISTING ACCESS NAME sowie für VERIFY ACCESS NAME wurde LP02 angegeben.

```
BACKUP DIRECTORY/VERIFY BACKUP - FOREGROUND/BACKGROUND
            DIRECTORY PATHNAME:          DATA1. ZENT
       SEQUENTIAL ACCESS NAME:           CMD48. DIRX8
         CONTROL ACCESS NAME:
          LISTING ACCESS NAME:           LP02
                      OPTIONS:
         KIF COMPRESS? (Y/N):
               VERIFY? (Y/N):            YES
          FOREGROUND? (Y/N):             YES

          VERIFY ACCESS NAME:            LP 02


   BACKUP DIRECTORY                      10:15:00 MONDAY,  JAN 04,  1982.

ORIGINAL SOURCE:                   DATA1. ZENT
ORIGINAL OUTPUT FILE:              CMD48. DIRX8
ORIGINAL OPTIONS:                  ALIASES, MULTI
CONTROL FILE:
LIST FILE:                         LP02

   **    ICBLCP
   **    IMASGP
   **    DC$DIO
   **          END OF DIRECTORY  DATA1. ZENT
******   BACKUP DIRECTORY COMPLETED


   VERIFY BACKUP                         10:15:12 MONDAY,  JAN 04,  1982.

   **          DIRECTORY DATA1. ZENT
   **    ICBLCP
   **    IMASGP
   **    DC$DIO
   **          END OF DIRECTORY DATA1. ZENT

******   VERIFY BACKUP COMPLETED
```

3.9.4 Wiederherstellen eines Directory (RD Kommando)

Das RD (Restore Directory) Kommando kopiert ein Set of Files einer mittels BD (Backup Directory) erstellten Sicherungskopie wieder zurück. Die Sicherungskopie kann sich auf eine oder mehrere Platten oder Magnetbänder erstrecken. Dabei ist zu beachten, daß beim Zurückkopieren die Sicherungskopien genau wieder in der Reihenfolge eingelegt werden, wie sie mittels des BD Kommandos erstellt wurden.

Bei dem Parameter SEQUENTIAL ACCESS NAME ist die Angabe einer Liste von Geräten möglich. Die Geräte werden in der Reihenfolge angesprochen, wie sie in der Liste angeführt wurden.

Dieses Kommando kann sowohl im Hintergrund als auch im Vordergrund abgearbeitet werden. Außerdem ist es möglich, sofort einen Verifizierungslauf anzuschließen.

3.9.4.1 RD Kommando Format

```
RD
RESTORE DIRECTORY
SEQUENTIAL ACCESS NAME: <acnm> @
DIRECTORY PATHNAME:     <acnm> @
CONTROL ACCESS NAME:    [ acnm] @
LISTING ACCESS NAME:    [ acnm] @
OPTIONS: [ (string)]
KIF COMPRESS? (Y/N): [ <yes/no>]
VERIFY (Y/N): <yes>
FOREGROUND (Y/N): <yes/no>
VERIFY ACCESS NAME:
```

3.9.4.2 Benutzerantworten zum Kommando RD

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
SEQUENTIAL ACCESS NAME:	R	Der Gerätename oder der Pfadname einer sequentiellen Datei, von welcher die Informationen wieder in das Directory übertragen werden sollen. Diese Information muß zuvor auf dieses Gerät bzw. auf diese Datei mittels eines Backup Directory Kommandos geschrieben worden sein. Gibt der Benutzer DSxx ein, erscheint am Bildschirm folgende Systemmeldung: PATHNAME OF BACKUP FILE: PATHNAME: [acnm]
DIRECTORY PATHNAME:	R	Der Pfadname des Directory oder der Datei, in welche die Information übertragen werden soll.
CONTROL ACCES NAME:	O	Der Name des Gerätes oder der Pfadname der Datei, von welcher die Anweisungen zur Kontrolle der Sicherungsprozedur kommen. Benutzer, die keinen CONTROL ACCESS NAME eingeben, müssen sowohl einen Directory Pfadname als auch einen Sequential Access Name angeben.

Vorsicht

Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei CONTROL ACCESS NAME 'ME' unzulässig. Im Vordergrund ist die Eingabe von 'ME' erlaubt.

LISTING ACCESS NAME: O

Der Gerätename oder Pfadname der Datei, in welche eine Zusammenfassung der Ergebnisse der Restore Operation ausgegeben werden kann. Standardwert ist der Terminal Local File des Benutzerterminals. Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei LISTING ACCESS NAME 'ME' unzulässig.

OPTIONS: O

Zur Überwachung der Restore Operation können folgende Options eingegeben werden, wobei ADD die Standardoption ist:

OPTION:	Zweck
ADD:	Dateien und Alternativnamen werden nur dann kopiert, wenn eine Datei oder ein Alternativname des gleichen Namens in dem 'Set of Files' der Ausgabe nicht existiert.
REPLACE:	Dateien werden selbst dann kopiert, wenn ein gleicher Name schon im 'Set of Files' der Ausgabe vorhanden ist. Diese bereits existierenden Dateien werden gelöscht, bevor die Quelldateien oder Dateien übertragen werden. Alternativnamen werden nicht ersetzt.
MULTI:	Wird dann angegeben, wenn sich die Quellinformation über mehr als ein Magnetband erstreckt.
NOMULTI:	(Standardwert) wenn sich die Quellinformationen nicht über mehrere Magnetbänder erstrecken.
DATE:	Erlaubt den Kopierprozess auf Dateien zu beschränken, deren letztes Updatedatum größer oder gleich dem angegebenen Datum ist.

Die Optionen ALIASES, NOALIASES, NODATE, REWIND, NOREWIND, UNLOAD, NOUNLOAD werden im Paragraph 3.10 beschrieben.

KIF COMPRESS? (Y/N)	O	Die Eingabe von Y(YES) bewirkt, daß eine oder mehrere Key Indexed Dateien, die unter dem bei INPUT PATHNAME eingegebenen Directory liegen, bzw. dort selbst angegeben sind, auf dem Zielmedium (OUTPUT PATHNAME) nur noch so groß engelegt werden, wie dies zur Aufnahme des aktuellen Datenbestandes erforderlich ist. Bei Eingabe von N(NO) jedoch bleibt die ursprüngliche Größe der Key Indexed Datei(en) erhalten.
VERIFY (Y/N):	R	Bei Eingabe von Y (YES) wird anschließend sofort ein Verifizierungslauf durchgeführt. Bei Eingabe von N (NO) wird nur das RD-Kommando abgearbeitet.
FOREGROUND (Y/N):	R	Die Eingabe von Y (YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N (NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.
VERIFY ACCESS NAME	O	Der Name eines Gerätes oder einer Datei, auf welche die Ergebnisse des Vergleichs ausgegeben werden sollen. Vorgabewert ist der Terminal Local File des Benutzerterminals.

Vorsicht:

Soll der Verifizierungslauf im Hintergrund laufen, so darf das Kontrollisting weder auf dem Bildschirm noch im Terminal Local File erscheinen.

Es ist darauf zu achten, daß die Angabe bei VERIFY ACCESS NAME mit der bei LISTING ACCESS NAME nicht identisch ist, sofern es sich um Pfadnamen von Dateien handelt, sonst wird das Kontrollisting des Sicherungsvorgangs überschrieben.

Anmerkung:

Falls bei der Systemmeldung OPTIONS als Benutzerantwort DATE eingegeben wurde, so erscheinen neben der Meldung "CD: DATE OF OLDEST FILE" noch folgende Systemmeldungen:

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
YEAR	R	Eingabe des Jahres in dezimaler oder hexadezimaler Form.
MONTH	R	Eingabe des Monats in dezimaler oder hexadezimaler Form.
DAY	R	Eingabe des Tages in dezimaler oder hexadezimaler Form.
HOUR	O	Eingabe der Stunde in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Stunde 0 angenommen.
MINUTE	O	Eingabe der Minute(n) in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Minute(n) 0 angenommen.

3.9.4.3 Beispiel zum RD Kommando

Das folgende Beispiel zeigt, wie man mittels eines RD Kommandos ein Directory wiederherstellen kann, das durch ein Backup Directory (BD) Kommando gesichert wurde (wie schon früher in diesem Kapitel gezeigt).

```
( ) RD
  RESTORE DIRECTORY
  SEQUENTIAL ACCESS NAME: .SEQTST
  DIRECTORY PATHNAME: .DIRTEST
  CONTROL ACCESS NAME:
  LISTING ACCESS NAME:  ME
  OPTIONS:  ADD,NOMULTI
  KIF COMPRESS? (Y/N): N
  VERIFY (Y/N):  NO
  FOREGROUND (Y/N):  YES

  RESTORE DIRECTORY                          03:57:51 SUNDAY, JUL 16, 1978.

  ORIGINAL SOURCE: .SEQTST
  ORIGINAL DESTINATION: .DIRTEST
  ORIGINAL OPTIONS: ADD, NOMULTI
  LIST FILE: ST02

      FILE1
      FILE3
      FILE2
      FILE4
           END OF DIRECTORY .DIRTEST

     RESTORE DIRECTORY COMPLETED
```

3.9.5 Prüfen einer Wiederherstellung oder einer Sicherung (VB-Kommando)

Das VB (Verify Backup) Kommando wird verwendet, um das Ergebnis eines BD (Backup Directory) Vorganges oder eines RD (Restore Directory) Vorganges zu überprüfen. Mittels des VB Kommandos kann ein Set of Files einer sequentiellen Datei, eines oder mehrere Magnetbänder oder Platten mit einem Set of Files eines Directory einer Platte verglichen werden oder umgekehrt. Das VB-Kommando findet diese Übereinstimmungen durch Vergleichen des Dateityps, Verwendung der Datei, Dateiname und Dateiinhalte der Dateien auf einem Gerät oder der Dateien, die auf eine durch den Benutzer bezeichnete Datei kopiert wurden. Besteht die zu verifizierende Sicherungskopie aus mehreren Magnetbändern oder Platten, so sind diese in der gleichen Reihenfolge einzulegen, wie beim eigentlichen Kopiervorgang. Befinden sich die Sicherungskopien auf verschiedenen Geräten, so sind diese Geräte in einer Liste anzugeben, wobei wieder die richtige Reihenfolge einzuhalten ist. Dieses Kommando kann sowohl im Vordergrund als auch im Hintergrund abgearbeitet werden.

Anmerkung

Das VB-Kommando verifiziert automatisch das nächste Directory, falls sich das zu verifizierende Directory auf einem Magnetband befindet. Aus diesem Grund muß das VB-Kommando für jedes neue Directory neu gestartet werden.

3.9.5.1 Format des VB Kommandos

```
VB
VERIFY BACKUP
SEQUENTIAL ACCESS NAME: <acnm>@
DIRECTORY PATHNAME: <acnm>@
CONTROL ACCESS NAME: [ acnm ] @
LISTING ACCESS NAME: [ acnm] @
OPTIONS : [ (string) ]
KIF COMPRESS? (Y/N): [ <yes/no>]
FOREGROUND (Y/N): <yes/no>]
```

3.9.5.2 Benutzerantworten zum VB Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
SEQUENTIAL ACCESS NAME:	R	Der Gerätename oder der Pfadname einer sequentiellen Datei, welche die zur Sicherung vorgesehene Kopie enthält. Das VB Kommando vergleicht den Set of Files bezüglich oben genannter Faktoren mit dem Set of Files, welcher durch den Pfadnamen des Directories angegeben wurde (siehe auch die folgende Systemmeldung). Gibt der Benutzer DSxx als SEQUENTIAL ACCESS NAME ein, so erscheint folgende Systemmeldung am Bildschirm: PATHNAME OF BACKUP FILE:
DIRECTORY PATHNAME:	R	Ein Pfadname, der das Directory auf dem höchsten Level angibt, welches den Set of Files enthält. Dieser Set of Files wird mit dem Set of Files verglichen, der der Sicherung dient und bei der vorhergehenden Systemanfrage eingegeben wurde.
CONTROL ACCESS NAME:	O	Der Name des Gerätes oder der Pfadname der Datei, von welcher die Anweisungen zur Kontrolle der Sicherungsprozedur kommen. Benutzer, die keinen Controll Access Name eingeben, müssen sowohl einen Directory Pfadname als auch einen Sequential Access Name angeben. Vorsicht Wenn das Kommando im Hintergrund abgearbeitet werden soll, ist bei CONTROL ACCESS NAME "ME" unzulässig. Im Vordergrund ist die Eingabe von "ME" erlaubt.
LISTING ACCESS NAME:	O	Der Gerätename oder Pfadname der Datei, auf welcher eine Ergebniszusammenfassung der Verify Operation ausgegeben werden soll. Standardwert ist hier der Terminal Local File des Benutzerterminals.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
OPTIONS:	O	Erlaubt den Kopierprozeß auf Dateien zu beschränken, deren letztes Updatedatum größer oder gleich dem angegebenen Datum ist.

Anmerkung:

Falls bei der Systemmeldung OPTIONS als Benutzerantwort DATE eingegeben wurde, so erscheinen neben der Meldung "CD: DATE OF OLDEST FILE" noch folgende Systemmeldungen:

YEAR	R	Eingabe des Jahres in dezimaler oder hexadezimaler Form.
MONTH	R	Eingabe des Monats in dezimaler oder hexadezimaler Form.
DAY	R	Eingabe des Tages in dezimaler oder hexadezimaler Form.
HOUR	O	Eingabe der Stunde in dezimaler und hexadezimaler Form. Bei keiner Eingabe wird als Stunde 0 angenommen.
MINUTE	O	Eingabe der Minute(n) in dezimaler oder hexadezimaler Form. Bei keiner Eingabe wird als Minute(n) 0 angenommen.
FOREGROUND (Y/N):	R	Die Eingabe von Y(YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll. Die Eingabe von N (NO) bedeutet, daß das Kommando im Hintergrund abgearbeitet werden soll.

3.9.5.3 Beispiel zum Kommando VB

Das folgende Beispiel zeigt die Eingabe eines VB Kommandos, um eine Sicherungsprozedur, wie sie schon vorher in diesem Kapitel für das BD (Backup Directory) Kommando beschrieben wurde, zu prüfen.

```
( ) VB
  VERIFY BACKUP
  SEQUENTIAL ACCESS NAME: .SEQTST
  DIRECTORY PATHNAME: .DIRTEST
  LISTING ACCESS NAME: ME
  CONTROL ACCESS NAME:
  OPTIONS:
  FOREGROUND (Y/N): YES
  VERIFY BACKUP                                  05:27:49 SUNDAY, JUL 16, 1978.

        DIRECTORY .DIRTEST
    FILE1
    FILE3
    FILE2
    FILE4
        END OF DIRECTORY .DIRTEST

    VERIFY BACKUP COMPLETED
```

3.10 Kontrolldateien

Das CD (Copy Directory) Kommando, das VC (Verify Copy) Kommando sowie das BD (Backup Directory) Kommando kopieren gewöhnlich den ganzen Inhalt (außer die Dateien S$PRINT, S$DIAG, .S$ROLLA und .S$CRASH) einer Directory Datei, es sei denn, daß Anweisungen die Directories, Dateien oder Alternativnamen der Dateien, die kopiert werden sollen, einschränken. Anweisungen sind immer in Kontrolldateien enthalten. Options können in Kontrolldateien durch Option Anweisungen erscheinen, als auch auf die Meldung OPTIONS dem System übergeben werden.

Kontrolldateien sind sequentielle Dateien, welche Anweisungen und Parameter dafür enthalten. Kontrolldateien werden üblicherweise durch den Text Editor erstellt, obwohl sie auch unmittelbar vom Bildschirm aus eingegeben werden können. Bild 3-3 zeigt ein Beispiel einer Kontrolldatei.

Wie Bild 3-3 zeigt, können Anweisungen überall in einer Zeile einer Kontrolldatei stehen, sie können auch abgekürzt werden; hinter einem Ausrufezeichen kann ein Kommentar geschrieben werden.

Tabelle 3-1 beschreibt die Anweisungen, die in Kontrolldateien benutzt werden können.

ANMERKUNG:

Synonyme können nur in der MOVE Anweisung für Pfadnamen oder Teile eines Pfadnamens verwendet werden.

```
OPTION REPL
MOVE VOL1. MYFILES
EX LIST, CONTROL
O ADD
MOVE VOL1. HISFILES
IN    A
IN    B
IN    C
INCL  X
I     PROG
END
```

Bild 3-3 Beispiel einer Kontrolldatei

Tabelle 3-1 Kontrolldatei Anweisungen

MOVE <Quellpfadname>, <Zielpfadname>
 	
MOVE <mastpfadname>, <copypfadname>
	Bei einem CD Kommando gibt die MOVE Anweisung den Pfadnamen (oder dessen Synonym) einer Quelldatei oder eines Directories (Quellpfadname), welche(s) in ein Zieldirectory (Zielpfadname) kopiert werden soll. So kann zum Beispiel eine MOVE Anweisung in einer Kontrolldatei wie folgt aussehen:
	
	MOVE VOL1. DIR1. FILE1, VOL3. DIRC
	
	Bei einem BD Kommando gibt eine MOVE Anweisung einen Pfadnamen einer Quelldatei oder eines Directories an, welche(s) in eine sequentielle Zieldatei oder auf ein Magnetband (Copy Access Name) übertragen werden soll.
	
	Beispiel: MOVE VOL1. DIR1. MT01

ANMERKUNG:

Jene Dateien oder Directories, welche auf Systemanfragen für Quelle und Ziel bei den Kommandos angegeben sind, werden vor den Dateien und Directories kopiert, die in der MOVE Anweisung angegeben wurden.

Eine durch ein VC Kommando aufgerufene MOVE Anweisung gibt den Pfadnamen eines Master File oder Directory (mastpfadname) an, welche(s) mit einer Copy Datei oder einem Copy Directory (copypfadname) verglichen werden soll.

Tabelle 3-1 Kontrolldateianweisung (Forts.)

Anweisung:	Zweck:
MOVE<Zielpfadname>	Ein neues Zieldirectory wurde definiert, aber die vorige Quelle noch benutzt.
MOVE<Quellpfadname >	Eine neue Quelle wurde definiert, aber das alte Ziel wird (weiter) benutzt.
MOVE<Quellpfadname>, <Zielpfadname>	Der Pfadname des Zieles soll auf die nächste Zeile der Kontrolldatei gerückt werden.
EXCLUDE <Dateiname>	Ein Dateiname besteht aus einem bis acht alphanumerischen Zeichen und enthält keine Punkte. Zulässige Dateinamen sind entweder Directories oder Dateien, die im Ausdruck eines List Directory aufgeführt werden. Jede EXCLUDE Anweisung benennt eine oder mehrere Datenfiles oder Subdirectories in dem zu kopierenden Directory, welche nicht kopiert oder geprüft werden sollen. Werden mehr als ein Dateiname in der EXCLUDE Anweisung angegeben, so müssen sie durch Kommata getrennt werden. Zum Beispiel bestimmt die folgende EXCLUDE Anweisung, daß alle Dateien, außer FILE1, FILE2, und FILE3 kopiert oder geprüft werden: EXCLUDE FILE1, FILE2, FILE3 Maximal können 50 Untergliederungen eines Directory durch ein oder mehrere EXCLUDE Anweisungen ausgeschlossen werden. Sich auf eine MOVE Anweisung beziehende EXCLUDE Anweisung muß unmittelbar nach der MOVE Anweisung in der Kontrolldatei folgen. Eine EXCLUDE Anweisung kann sich nur dann auf eine MOVE Anweisung beziehen, wenn jener keine INCLUDE Anweisung zugeordnet ist.

Tabelle 3-1 Kontrolldateianweisung (Forts.)

Anweisung:	Zweck:
INCLUDE Dateiname	Ein Dateiname besteht aus einem bis acht alphanumerischen Zeichen und enthält keine Punkte. Zulässige Dateinamen sind entweder Directories oder Dateien, die im Ausdruck eines List Directory aufgeführt werden. Die INCLUDE Anweisung gibt an, daß keine Dateien oder Subdirectories des genannten Directory kopiert oder geprüft werden, außer jenen, die in der INCLUDE Anweisung aufgeführt sind. Eine INCLUDE Anweisung nach einer MOVE Anweisung ist nicht zulässig, wenn schon eine EXCLUDE Anweisung der MOVE Anweisung folgt. In einer INCLUDE Anweisung angeführte Dateinamen sind durch Kommata zu trennen. Beispielsweise gibt die folgende INCLUDE Anweisung an, daß nur FILE1, FILE2 und FILE14 kopiert oder geprüft werden sollen: INCLUDE FILE1, FILE2, FILE14 Maximal 50 Untergliederungen eines Directory können in einer oder mehreren INCLUDE Anweisungen aufgeführt werden. Eine zu einer MOVE Anweisung gehörige INCLUDE Anweisung muß in der Kontrolldatei unmittelbar der MOVE Anweisung folgen. Müssen mehr als 50 Bereiche eines Directory bewegt werden, können dazu mehrere MOVE Anweisungen mit gleichen Quell- und Zielpfadnamen angegeben werden.
OPTION Option	Optionen, die in der OPTION Anweisung angegeben wurden, kontrollieren den Austausch der Datei und das Kopieren der Alternativnamen. OPTION Anweisungen können überall in einer Kontrolldatei außer zwischen EXCLUDE Anweisungen vorkommen. Folgende Optionen können in der OPTION Anweisung auftreten:

Option:	Zweck:
ADD	Dateien werden kopiert, außer die Datei existiert schon mit gleichem Namen und auf gleichem Level im Zieldirectory. Die ADD Option wird vom VC (Verify Copy) Kommando ignoriert.

Tabelle 3-1 Kontrolldateianweisung (Forts.)

Option:	Zweck:
REPLACE	Dateien werden auch dann kopiert, wenn eine Datei gleichen Namens und auf gleichem Level schon im Zieldirectory vorhanden ist. Diese bereits existierende Datei wird gelöscht und durch die zu kopierende Datei ersetzt. Die REPLACE Anweisung wird durch das VC (Verify Copy) Kommando ignoriert.
ALIASES	Alle Alternativnamen einer zu kopierenden oder zu prüfenden Datei werden kopiert oder geprüft, es sei denn, der Alternativname existiert schon auf dem gleichen Level im Zieldirectory.
NOALIASES	Es werden keine Alternativnamen kopiert oder geprüft. Werden in einer OPTION Anweisung keine Optionen angegeben, so werden die vorhergehenden Optionen weiter benutzt und eine Warnung (auf die Ausgabedatei) ausgedruckt.
MULTI	Das gesicherte oder zu sichernde Directory erstreckt sich über mehrere Magnetbänder.
NOMULTI	Das Directory erstreckt sich nicht über mehrere Bänder (oder ist auf eine sequentielle Datei gesichert).
DATE	Beschränkt die Datensicherung bzw. den Kopiervorgang auf Dateien, deren letztes Update Datum größer oder gleich dem angegebenen Datum ist. Folgendes Format ist erforderlich: YEAR: <int> MONTH: <int> DAY: <int> HOUR: [int] MINUTE: [int]
BLOCK	Eingabe von Block bewirkt eine Aufteilung der Sätze in Blöcke mit bestimmter Größe. Dazu notwendig ist die Angabe des Block-Faktors: BLOCKING FACTOR: [int] Jeder Block wird als physischer Satz geschrieben. Der ursprüngliche Wert ist 9600 Bytes.

NOBLOCK	Die Eingabe bewirkt, daß jeder Satz gesondert ohne Blockung geschrieben wird.
NODATE	Eingabe von NODATE bedeuted kein Datumszwang bei der Datensicherung.
REWIND	Auf Grund der Angabe wird das Band noch vor Ausführung der Datensicherungsprozedur zurückgespult.
NOREWIND	Das Band wird vor Ausführung der Datensicherungsprozedur nicht zurück gespult.
UNLOAD	Zeigt an: Das Band wird nach Verarbeitung aufgeladen.
NOUNLOAD	Das Band wird nach Verarbeitung nicht entladen.
SPRL	Eine sequentielle Datei wird in eine neue physikalische Satzlänge umgesetzt. Das Betriebssystem TAXO zeigt am Display folgenden Hinweis: PHYSICAL RECORD LENGTH: <int>
NOSPRL	Die sequentielle Datei bleibt in ihrer physikalischen Satzgröße erhalten.
RPRL	Eine relative Datei wird in eine neue physikalische Satzgröße umgesetzt. Das Betriebssystem TAXO zeigt am Terminal den Hinweis: PHYSICAL RECORD LENGTH: <int>
NORPRL	Eine relative Datei bleibt in ihrer physikalischen Satzgröße enthalten.

* Optionen können abgekürzt werden, solange sie dadurch eindeutig bleiben. Z.B. könnte A sowohl für ADD als auch für ALIASES stehen. Dies ist zweideutig und sollte daher durch AD oder AL ersetzt werden.

4 PROGRAMMAUSFÜHRUNG UND KONTROLLE

4.1 Allgemeines

Unter dem Betriebssystem TAXO ausgeführte Programme heißen Tasks. Alle Programme werden zu Tasks, wenn sie durch ein XT (Execute Task) oder XTS (Execute Task and Suspend SCI) Kommando aktiviert wurden. Das XT Kommando wird gebraucht umTasks zu aktivieren, die nicht mit Terminals des Systems kommunizieren. Das XTS Kommando benutzt man dagegen zum Aufruf von Programmen, die mit Terminals kommunizieren. Aktivierte Programme bekommen durch TAXO eine Laufzeit-ID (RUNTIME ID) zugeordnet, es sei denn, sie sind nicht replikatibel.

Dieses Kapitel beschreibt das XT und das XTS Kommando, welche Tasks aktivieren. Dieses Kapitel beschreibt auch das STS (Show Task Status) Kommando, durch welches man sich den Status der Tasks anzeigen lassen kann und das KT (Kill Task) Kommando, mit dem man Tasks abbrechen kann.

4.2 Ausführen nichtinteraktiver Tasks (XT Kommando)

Um ein Programm aufzurufen, das nicht mit dem Terminal des Benutzers kommuniziert, wird das XT (Execute Task) Kommando eingegeben. Dem Programm können zwei 16-Bit Worte als Antwort auf die PARM1: und PARM2: Systemmeldungen übergeben werden. Jedes durch TAXO aufgerufene Programm bekommt vom Betriebssystem automatisch eine Laufzeit-ID zugeordnet. TAXO zeigt die Laufzeit-ID am Bildschirm an, nachdem das Programm erfolgreich gestartet wurde. Der Gebrauch dieses Kommandos verhindert, daß die Task auf Event Characters zugreifen kann. Ein mit XT gestartetes Programm kann im allgemeinen nicht auf Synonyme zugreifen (notwendig z. B. bei COBOL Programmen; siehe hierzu XCT, XCP im COBOL Handbuch.

4.2.1 Format des XT Kommandos

```
XT
EXECUTE TASK
PROGRAM FILE OR LUNO: <string> @
TASK NAME OR ID: <string>
PARM1: <int>
PARM2: <int>
STATION ID: <string>
```

4.2.2 Benutzerantworten zum XT Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PROGRAM FILE:	R	Ein Pfadname oder eine LUNO der Programmdatei, in welcher das Programm steht. Der Wert 0 bezeichnet die Programmdatei des Systems.
TASK NAME OR ID:	R	Entweder den Namen oder die ID-Nummer, unter welcher das Programm angelegt wurde
PARM1: PARM2:	R R	PARM1: und PARM2: sind hexadezimale oder dezimale Zahlen zwischen 0 und 65535. Sie bestimmen den Wert, der dem Programm übergeben wird.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
STATION ID:	R	Die Nummer (02 nicht ST02) der Station, mit welcher die auszuführende Task verbunden ist. 0 (Null) oder ME bezeichnet den Bildschirm des Benutzers.

4.2.3 Beispiel zum XT Kommando

Das folgende Beispiel zeigt, wie mit Hilfe des XT Kommandos eine Task mit der hexadezimalen ID-Nummer A5 von der System Programm Datei aufgerufen und diese auszuführende Task mit der Station des Benutzers verbunden wird. Wie das Beispiel zeigt, gibt TAXO die Laufzeit-ID aus, welche automatisch zugeordnet wird, sobald das Programm aktiviert wird.

```
( ) XT
   EXECUTE TASK
   PROGRAM FILE OR LUNO: .S$PROGA
   TASK NAME OR ID: >A5
   PARM1: 0
   PARM2: 0
   STATION ID: ME

   RUNTIME TASK ID = >3A
( )
```

4.3 Ausführen interaktiver Tasks (XTS Kommando)

Das Kommando XTS (Execute Task and Suspend SCI) aktiviert ein interaktives Programm. Das XTS Kommando schaltet automatisch den SCI aus, um eine Störung der Ausführung des interaktiven Programmes zu verhindern. Würde der SCI nicht abgeschaltet sein, so würde dieser weiterhin Daten, die vom Bildschirm eingegeben werden, interpretieren, so als würden sie ihm gelten. Ein Fehler wäre die Folge. Dieses Kommando wird auch gebraucht, um Event Characters für eine Task nutzen zu können, statt für den SCI. Ein mit XTS gestartetes Programm kann im allgemeinen nicht auf Synonyme zugreifen (notwendig z.B. bei COBOL Programmen; siehe hierzu z.B. XCTF, XCPF im COBOL Handbuch)

ANMERKUNG:

Falls auf die Systemfrage STATION: eine andere als die eingebende Station bezeichnet wird, wird der SCI dennoch an der eingebenden Station abgeschaltet. Ist die Task beendet, wird der SCI dort wieder reaktiviert.

4.3.1 Format des XTS Kommandos

```
XTS
EXECUTE TASK AND SUSPEND SCI
PROGRAM FILE OR LUNO: <string> @
TASK NAME OR ID: <string>
PARM1: <int>
PARM2: <int>
STATION ID: <string>
```

4.3.2 Benutzerantworten zum XTS Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PROGRAM FILE OR LUNO:	R	Ein Pfadname oder eine LUNO jener Programmdatei, in der das Programm abgelegt ist. Der Wert 0 bezeichnet die System Programm Datei.
TASK NAME OR ID:	R	Entweder der Name oder die ID Nummer, unter welcher das Programm angelegt wurde.
PARM1: PARM2:	R R	Hexadezimale oder dezimale Zahlen zwischen 0 und 65535, die dem Programm zugewiesene Werte angeben.
STATION ID:	R	Die Nummer (02, nicht ST02) derjenigen Station, die der auszuführenden Task zugeordnet werden soll. 0 gibt die Benutzerstation an.

4.3.3 Beispiel zum XTS Kommando

Nachfolgendes Beispiel zeigt das Starten einer interaktiven Task mit der zugeordneten ID $A5_{16}$ mittels eines XTS Kommandos, wobei die auszuführende Task mit der Station des Benutzers verbunden werden soll. Wie das Beispiel zeigt, listet TAXO die Laufzeit-ID (welche durch TAXO automatisch zugeordnet wird) auf, sobald das Programm erfolgreich gestartet wurde.

```
( ) XTS
   EXECUTING TASK AND SUSPEND SCI
   PROGRAM FILE OR LUNO: .S$PROG
   TASK NAME OR ID: >A5
   PARM1: 0
   PARM2: 0
   STATION ID: ME
   RUNTIME TASK ID = >38
( )
```

4.4 Ausführen und Anhalten einer Task (EXECUTE AND HALT TASK - XHT)

Das Kommando XHT lädt eine Task in den Speicher und suspendiert diese, um ihre Ausführung unter Debug zu ermöglichen. Typisches Vorgehen: Der Benutzer bringt die Task, die unter Debug laufen soll mit XHT in den Speicher, startet für die Task den Debugger (einschließlich der Haltepunkte - Breakpoints) und aktiviert dann die Task, indem er die Kommandos Aktivieren einer Task (Activate Task-AT) oder Fortsetzen der Task (Resume Task-RT) verwendet. Dieses Kommando besitzt folgendes Format:

```
          XHT
   EXECUTE AND HALT TASK
     PROGRAM FILE OR LUNO:    <string>
           TASK NAME OR ID:   <string>
                  PARM1:      <int>
                              0
                  PARM2:      <int>
                              0
              STATION ID:     <string>
                              ME
```

4.5 Anzeigen des Status von Tasks (STS Kommando)

Das STS (Show Task Status) Kommando gibt eine Statusbeschreibung einer speziellen Task oder aller Tasks, die gerade im System ablaufen; Tabelle 4-1 beschreibt die Information, welche für jede Task unter den, durch ein STS Kommando erzeugten Ausgabeüberschriften gegeben wird. Während der Ausführung des STS Kommandos kann eine Task im Speicher neu bereitgestellt werden, wodurch die Task zweifach in der STS Ausgabe erscheint. Das geschieht nicht oft und bedeutet kein Problem.

Tabelle 4-1 Durch ein STS Kommando erzeugte Informationen über den Task Status

Überschrift	Beschreibung
BID ID	Die ID der Task, deren Status angezeigt wird.
RUN ID	Die Laufzeit ID, die der Task beim Start durch TAXO zugewiesen wurde.
STATION	Die Nummer der Station, mit der die auszuführende Task verbunden ist.
STATE	Ein Code, der den Status der Task wie folgt beschreibt:

Code	Bedeutung
00	Aktive Task
01	Aktive Task
04	Task beendet
05	Task im Time delay
06	Suspendierung einer Task
07	gerade ablaufende Task
08	Task erwartet Zeicheneingabe
09	Task erwartet Beendigung der Ein-/Ausgabe
0A	Task erwartet die Zuordnung eines I/O-Gerätes
0B	Task wartet auf Dienste von Plattendateidienstprogrammen
0C	Task wartet auf Compress services
0D	Task wartet auf Dienste des File Managements
0E	Task wartet auf Dienste des Overlay Loaders
0F	Task wartet auf Dienste des Task Loaders
11	Task wartet auf Dienste des Platten Managements
12	Task wartet auf Dienste des Band Managements
13	Task erwartet Dienste des System Overlay Loaders
14	Task wartet auf den durch die Task gesteuerten Supervisor Call Processor
15	Task erwartet Dienste des Get Memory
17	Task suspendiert, um eine gerufene Task aktivieren zu können
18	Task erwartet diagnostische Task Dienste
19	Task erwartet die Beendigung einer I/O
1A	Task erwartet Dienste des Speicher Managements
1B	Task ist für roll-out vorgesehen, sobald die geforderte I/O beendet ist.
1C	Task wurde ausgelagert
1D	Suspendiert wegen initiate I/O
1E	Suspendiert wegen eines gesperrten Directories
1F	Suspendiert wegen Task Management Directory Buffer
24	Task suspendiert von Eingabe aus der Warteschlange
FF	Dummy Task Status

Tabelle 4-1 Durch ein STS Kommando erzeugte Informationen über den Task Status (Fortsetzung)

Überschrift	Beschreibung
PRIORITY:	Ein Code der die Priorität einer Task wie folgt angibt:

Code	Bedeutung
00	Die Task ist eine System Task (höchste Priorität)
R1	Realzeit Priorität eins
.	
.	
.	
R127	Realzeit Priorität 127
01	Zweite Prioritätsstufe
02	Dritte Prioritätsstufe
03	Niedrigste Priorität
04	Unterschiedliche Prioritäten, deren Stufen durch TAXO abhängig von der Task vergeben werden. Es ist dies die für nichtprivilegierte interaktive Benutzertasks empfohlene Priorität.

FLAG 1 Ein Task Status Hinweisblock, in welchem Bits gesetzt sind, die die Charakteristika einer Task wie folgt anzeigen:

Bit	Task Charakteristik
0	System Task
1	Privilegierte Task
2	Speicherresidente Task
3	Die Task muß im Falle eines Fehlers eine Endebehandlung durchführen
4	zur Auslagerung vorgesehen
5	ausgelagert
6	Die Task ist vorgesehen, abgebrochen oder beendet zu werden.
7	Die Task sollte durch eine andere Task reaktiviert werden, während sie noch tätig war. Sobald sie endet, wird sie neu reaktiviert.
8	Die Task wurde durch eine andere Task aufgerufen und jene abgebrochen. Nach Beendigung der Task wird die abgebrochene Task reaktiviert.
9	Seriell wieder nutzbare Task
10	Task wird weiter abgearbeitet
11	Die Task wurde erstmals gestartet
12	Task allein lassen
13	Task wird durch einen anderen TSB kontrolliert
14	SCI Hinweis zur Prüfung der TSB Kette
15	Wiederholbare (replicatable) Task

Tabelle 4-1 Durch ein STS Kommando erzeugte Informationen über den Task Status (Fortsetzung)

Überschrift	Beschreibung
FLAG 2	Ein Task Status Hinweisblock, in welchem Bits gesetzt sind, die die Charakteristika einer Task wie folgt anzeigen:
	Bit — Task Charakteristik
	0 — Beim nächsten Start der Task soll sie suspendiert werden.
	1 — Die Task wird kontrolliert
	2 — Falls angegeben, sollen SVC Traps benützt werden.
	3 — SVC Schalter: bei 0 SVC Trap nehmen
	4 — Ausführung durch den Scheduler gestoppt
	5 — Ausführung durch trapped SVC gestoppt
	6 — Ausführung durch XOP 15, 15 gestoppt
	7 — dynamisches Prioritätenmanagement
	WP — Die Registeradresse des Workspace Pointer relativ zur Speicheradresse der Task
	PC — Die laufende Befehlsadresse der auszuführenden Task relativ zur Speicheradresse der Task

4.5.1 Format des STS Kommandos

STS
SHOW TASK STATUS
INSTALLED ID: [int]
OUTPUT ACCESS NAME: [acnm] @

4.5.2 Benutzerantworten beim STS Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INSTALLED ID:	O	Die zugewiesene Identifikationsnummer der Task, deren Status angezeigt oder aufgelistet werden soll. Wird nichts angegeben, so wird der Status aller aktiven Benutzertasks angezeigt.
OUTPUT ACCESS NAME:	O	Der Gerätename oder der Pfadname der Datei, auf welche TAXO die Statusinformationen ausgeben soll, die durch das STS Kommando erzeugt werden. Standard ist der Terminal Local File.

4.5.3 Beispiel zum STS Kommando

Das nachfolgende Beispiel zeigt, wie mittels Eingabe eines STS Kommandos der Status einer Task mit der zugewiesenen ID 20_{16} am Terminal des Benutzers angezeigt wird:

```
STS
SHOW TASK STATUS
   INSTALLED ID: >20
   OUTPUT ACCESS NAME: ME
```

BID ID	RUN ID	STATION	STATE	PRIORITY	FLAG1	FLAG2	WP	PC
20	1C	6	09	01	5001	0100	3B30	24C0
20	0D	1	17	01	5001	0100	3B30	5518
20	C2	4	09	01	5001	0100	3B30	24C0
20	62	7	05	01	5001	0100	3BBC	487C
20	48	9	09	01	5401	0100	3B30	24C0
20	2B	8	09	01	5401	0100	3B30	24C0
99	99	3	09	R80	5401	0100	0050	0100

4.6 Beenden von Tasks (KT Kommando)

Das Kommando KT (Kill Task) beendet eine aktive Task. Tasks, die eine Endebehandlung durchführen, können diese noch zuendeführen. TAXO zeigt den Status der Task am Terminal des Benutzers an, sobald sie beendet ist.

Tasks mit einem der folgenden Status werden eine Ende-Behandlung durchführen oder sofort abbrechen, nachdem ihre Beendigung gefordert wurde.

- 00 Task erwartet Zeitscheibenzuweisung
- 01 Task erwartet Zeitscheibenzuweisung
- 04 Task ist beendet
- 05 Task im Time Delay
- 06 Task uneingeschränkt suspendiert
- 08 Task erwartet Eingabe von Zeichen
- 09 Task wartet auf Beendigung der Ein-/Ausgabe
- 0A Task wartet auf Zuweisung eines Gerätes
- 0D Task wartet auf Hilfsdienste des File Managements
- 0F Task erwartet erste Aktivierung
- 17 Task wartet auf die Tätigkeit einer CO-Routine
- 19 Task erwartet die Beendigung irgendeiner Ein-/Ausgabe
- 24 Task beendet, erwartet Einreihung in die Warteschlange

Tasks, mit einem Status wie folgt, werden nicht beendet, solange sich ihr Status nicht ändert und einen der obigen Werte annimmt.

- 07 Im Augenblick in der Ausführung
- 0B Task erwartet Dienste der File Utility
- 0E Task erwartet Dienste des Overlay Loaders
- 11 Task erwartet Dienste des Disk Managements
- 13 Task erwartet Dienste des System Overlay Loaders
- 14 Task erwartet die Beendigung einer durch einen SVC Prozessor gesteuerten Warteschlange.
- 15 Task wartet auf die Beendigung eines GET Memory Auftrages
- 18 Task befindet sich gerade in einem Abschlußprozeß
- 1A Task fordert Speicher vom Speicher Management an (nur Betriebssystem)
- 1C Task wird gerade ausgerollt
- 1D Task hat einen Initiate I/O begonnen
- 1E Task wartet auf einen Zugriff zu einem gesperrten Directory (nur Betriebssystem)

4.6.1 Format des KT Kommandos

KT
KILL TASK
RUN ID: <int>
STATION NUMBER: <int>

4.6.2 Benutzerantworten zum KT Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
RUN ID:	R	Die RUNTIME ID, die der Task durch TAXO zu Beginn der Laufzeit zugeordnet wurde.
STATION NUMBER:	R	Die Nummer der Station, mit welcher die Task zu Laufzeitbeginn verbunden war. FF wird eingegeben, wenn die Task keiner Station zugeordnet war. 0 bezeichnet die Station des Benutzers.

4.6.3 Beispiel zum KT Kommando

Folgendes Beispiel zeigt den Einsatz des KT Kommandos, um eine Task mit der Runtime ID $5C_{16}$ und dem mit ihr verbundenen Terminal 4 (der Station mit der Gerätenummer ST04) abzubrechen. TAXO meldet sich bei Beendigung der Task und gibt eine Meldung aus, die besagt, daß die Task zur Abbruchzeit das Ende einer Ein-/Ausgabe erwartete (Task Status 09).

```
( ) KT
    KILL TASK
    RUN ID: >50
    STATION NUMBER: >4
    TASK STATE = >9
( )
```

4.7 Hard Break Sequence (Programmabbruch durch Tastendruck)

Beim Testen von Programmen oder bei fehlerhafter Programmausführung ist es manchmal wünschenswert, ein an einem Terminal laufendes Programm durch Tastendruck abzubrechen ("Hard Break"). Dies ist durch Drücken der Tastenfolge ("Sequence")

STOP
CONTROL X (gleichzeitiges Drücken der CONTROL Taste und des Buchstabens X)

auf der Tastatur des DTS möglich. Durch den Programmabbruch wird wieder der SCI aktiviert. Ein im Vordergrund laufendes Assembler Programm kann die Hard Break Sequence in einer End Action Routine selbst abfangen und geeignete Maßnahmen treffen (z.B. Dateien abschliessen, Wiederaufsetzen an der unterbrochenen Stelle u.ä.). In diesem Fall wird der SCI nicht aktiviert, sondern das Programm läuft weiter. Durch die Hard Break Sequence wird für die abgebrochene Task der neue Task Fehlercode >10 erzeugt und ins System Logbuch eingetragen.

WARNUNG

Durch die Hard Break Sequence kann ein Programm an jeder beliebigen Stelle abgebrochen werden. Dies kann zur Folge haben, daß das Programm inkonsistente Dateien u.ä. hinterläßt. Die Hard Break Sequence sollte deshalb nur mit größter Vorsicht angewandt werden.

Die Hard Break Sequence hat nur Wirkung auf Tasks, die vom SCI durch das Primitiv Kommando .BID gestartet wurden. Sie hat keine Wirkung auf den SCI selbst. Sie hat außerdem keine Wirkung, wenn der SCI in einem anderem Task Status als >17 (Suspended Awaiting Coroutine Completion) befindet.

Die Hard Break Sequence steht unter TAXO V 2.0.0 nur zur Verfügung, wenn in das DTS das neue - für TAXO V 2.0.0 freigegebene - Mikro eingebaut wurde. Ansonsten hat sie keine Wirkung.

5 KONTROLLE DES TERMINALZUGRIFFS

5.1 Allgemeines

Die Benutzer können den Zugriff zu TAXO kontrollieren, indem sie den Status von Terminals im System sowie die Benutzeridentifikation überprüfen. Die Kommandos, mit denen man den Terminalstatus und die Benutzeridentifikation überprüfen kann, lauten wie folgt:

- MTS Kommando (Modify Terminal Status, Verändern des Terminalstatus)
- LTS Kommando (List Terminal Status, Auflisten des Terminalstatus)
- AUI Kommando (Assign User ID, Zuweisen der Benutzeridentifikation)
- MUI Kommando (Modify User ID, Verändern der Benutzeridentifikation)
- DUI Kommando (Delete User ID, Löschen der Benutzeridentifikation)
- LUI Kommando (List User ID, Auflisten der Benutzeridentifikation)

5.2 Verändern des Terminalstatus - MTS Kommando

Der Benutzer kann das MTS Kommando (Modify Terminal Status) verwenden, um die folgenden Eigenschaften jedes Terminals im System zu überwachen:

- Terminals können in den Zustand EIN (ON) oder AUS (OFF) gesetzt werden. Terminals im Zustand ON stehen mit dem Systemkommando-Interpreter (SCI) in Verbindung und können benutzt werden, um Kommandos zur Kontrolle von TAXO-Operationen einzugeben. Terminals im Zustand OFF stehen nicht mit dem SCI in Verbindung, können somit nicht zur Kommandoeingabe verwendet werden, sondern nur als physikalische Ein-/Ausgabegeräte.

- Ein Terminal kann in die Lage versetzt werden, vom Benutzer ein Log-in zu verlangen, bevor es unter SCI arbeiten kann.

- DTS-Terminals können entweder im TTY- oder im VDT-Modus arbeiten. Im TTY-Modus gibt das Terminal jeweils nur eine Zeile auf den Bildschirm aus und akzeptiert jeweils nur eine Eingabezeile von der Tastatur. Im VDT-Modus kann jeweils der gesamte Bildschirm zur Ausgabe und Eingabe benutzt werden.

  ANMERKUNG:

  Wenn an einem Terminal, dessen Status gerade geändert wird, momentan mit dem SCI gearbeitet wird, kommen die neu definierten Eigenschaften nicht zur Wirkung, bis das Quit (Q-) Kommando eingegeben wird und der Benutzer am gleichen Terminal den SCI wieder aufruft.

5.2.1 Format des MTS Kommandos

```
MTS
MODIFY TERMINAL STATUS
TERMINAL NAME: <name>
NEW STATUS (ON/OFF): [ string ]
NEW MODE (TTY/VDT): [ string ]
LOGIN REQUIRED?: [ yes/no ]
USER PRIVILEGE CODE: [ integer ]
DEFAULT MODE (TTY/VDT): [ string ]
```

5.2.2 Benutzerantworten beim MTS Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
TERMINAL NAME:	R	Gerätename des Terminals, dessen Status geändert werden soll.
NEW STATUS (ON/OFF):	O	Eingabe "ON", wenn die Benutzer SCI-Kommandos vom Terminal aus eingeben dürfen. Eingabe "OFF", wenn verhindert werden soll, daß Benutzer das Terminal für SCI-Kommandos benutzen dürfen. Standardwert ist der jeweilige Status des Terminals.
NEW MODE (TTY/VDT):	O	Eingabe von "TTY", um den TTY-Modus zu erhalten, von "VDT" für den VDT-Modus. Standardwert ist der jeweilige Status des Terminals.
LOGIN REQUIRED?:	O	Eingabe "Y", um ein Login zu verlangen, bevor mit dem Terminal unter SCI gearbeitet werden kann, Eingabe von "N" (No/Nein), wenn es den Benutzern erlaubt ist, auch ohne Log-in vom Terminal aus auf den SCI zuzugreifen. Standardwert: Log-in erforderlich.
USER PRIVILEGE CODE:	O	Eingabe einer ganzen Zahl von 0 bis 7, um die Bevorrechtigungsstufe anzugeben, die von jedem Benutzer verlangt wird, der am Terminal ein Log-in durchführt. Benutzer mit niedrigen Bevorrechtigungsstufen (bestimmt durch die Bevorrechtigungsstufe, die zur Nummer der Benutzeridentifikation gehört) dürfen den SCI am Terminal nicht benutzen. Standardwert ist der derzeitige Wert für das betreffende Terminal.
DEFAULT MODE (TTY/VDT):	O	Eingabe von TTY oder VDT, um den Anfangsmodus zu bestimmen, der immer dann gilt, wenn der SCI vom Terminal aus aktiviert wird. Standardwert: VDT.

5.2.3 Beispiel zum MTS Kommando

Das folgende Beispiel zeigt die Eingabe eines MTS Kommandos, um die Eigenschaften des Terminals ST07 festzulegen und die Eingabe eines LTS Kommandos (List Terminal Status), um sich zu versichern, daß die gewählten Eigenschaften nun gültig sind.

```
() MTS
MODIFY TERMINAL STATUS
 TERMINAL NAME:          ST07
 NEW STATUS (ON/OFF):     ON
 NEW MODE (TTY/VDT):     TTY
 LOGIN REQUIRED?:       YES
 USER PRIVILEGE CODE:     2
 DEFAULT MODE (TTY/VDT):    VDT
()
LTS
LIST TERMINAL STATUS
 TERMINAL NAME:    ST07
 OUTPUT ACCESS NAME:    ME
 TERMINAL  USER  LI REQ'D  MODE  DEFAULT
   ST07   JKB044    YES     TTY    VDT


()
```

5.3 Auflisten des Terminalstatus - LTS Kommando

Man gibt das LTS Kommando (List Terminal Status) ein, um die zugewiesenen Eigenschaften eines bestimmten Terminals oder aller Terminals im System anzuzeigen oder aufzulisten und um die Benutzeridentifikation derjenigen Benutzer, die momentan das Terminal benutzen, anzuzeigen. Abb. 5-1 zeigt ein Beispiel für die Liste der Eigenschaften aller Terminals in einem System.

Tabelle 5-1 beschreibt die Status-Information, die in der Liste des LTS Kommandos unter jeder Zeilenüberschrift gegeben wird.

TERMINAL	USER	LI	REQ'D	MODE	DEFAULT
ST01	SYSTEM		NO	TTY	VDT
ST02			YES	VDT	VDT
ST03	WHT016		YES	VDT	VDT
ST04	CLN002		YES	VDT	VDT
ST05			YES	VDT	VDT
ST06	GPS021		YES	VDT	VDT
ST07	JKB044		YES	TTY	VDT
ST08	KAC009		YES	VDT	VDT
ST09			YES	VDT	VDT
ST10			YES	VDT	VDT
ST11			NO	TTY	VDT
ST12			NO	TTY	VDT
ST13			NO	TTY	VDT
ST14			NO	TTY	VDT
ST15			NO	TTY	VDT

Abb. 5-1 Ausgabe des LTS Kommandos

Tabelle 5-1 Ausgabe der Status-Information durch das LTS Kommando

Überschrift	Beschreibung
TERMINAL	Gerätename des oder der Terminals, deren Status aufgelistet werden.
USER	Benutzeridentifikation des Benutzers, der gerade am Terminal tätig ist (nach Log-in).
LI REQ'D	"Y" (yes) besagt, daß Benutzer ein Log-in am Terminal durchführen müssen, um mit dem SCI zu arbeiten. "N" (no) zeigt an, daß der SCI am Terminal aufgerufen werden kann, ohne daß sich der jeweilige Benutzer durch ein Log-in identifiziert hat.
MODE	"TTY" besagt, daß das Terminal momentan im TTY Modus arbeitet, "VDT" steht für den (gegenwärtigen) VDT Modus.
DEFAULT	"TTY" zeigt an, daß das Terminal immer im TTY Modus arbeitet, bis sein Status durch ein MTS Kommando geändert wird. "VDT" besagt, daß das Terminal immer im VDT Modus benutzt wird, bis sein Status durch ein MTS Kommando geändert wird.

5.3.1 Format des LTS Kommandos

LTS
LIST TERMINAL STATUS
TERMINAL NAME: [name]
OUTPUT ACCESS NAME: [acnm]

5.3.2 Benutzerantworten beim LTS Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
TERMINAL NAME:	O	Gerätename des Terminals, dessen Status angezeigt werden soll. Standardwert: Anzeige des Status aller Terminals.
OUTPUT ACCESS NAME:	O	Name des Gerätes, auf dem TAXO die Ergebnisse des LTS Kommandos anzeigen oder auflisten soll. Wahlweise kann ein Pfadname eingegeben werden, um eine Datei anzugeben, auf die TAXO die Ergebnisse des LTS Kommandos kopieren soll. Standardwert ist der Terminal Local File des betreffenden Terminals.

5.3.3 Beispiel zum LTS Kommando

Das folgende Beispiel zeigt die Eingabe eines LTS Kommandos, um den Status aller Terminals im System zu erhalten.

```
( ) LTS
    LIST TERMINAL STATUS
    TERMINAL NAME:
    OUTPUT ACCESS NAME: LP01
( )
```

5.4 Kontrolle der Benutzeridentifikation

TAXO Installationen, die ein Log-in am Terminal verlangen, müssen die Benutzer mit Benutzeridentifikationen und Passcodes versorgen. Die Benutzer identifizieren sich gegenüber dem Betriebssystem, indem sie ihre Benutzeridentifikation und ihren Passcode als Antworten auf Systemmeldungen während des Log-in Vorgangs eingeben. Wenn das Betriebssystem Identifikation und Passcode als gültig erkennt und die Bevorrechtigungsstufe größer oder gleich der ist, die am Terminal verlangt wird, läßt es den Benutzer am Terminal in der Bevorrechtigungsstufe arbeiten, die der Benutzeridentifikation vorher zugewiesen wurde. Die folgenden Abschnitte beschreiben, wie in TAXO Installationen Benutzeridentifikationen und Passcodes zugewiesen, geändert und aufgelistet (ASSIGN, MODIFY, LIST) werden können.

5.4.1 Zuweisung von Benutzeridentifikationen - AUI Kommando

Autorisierte Benutzer können Benutzeridentifikationen, Passcodes, Bevorrechtigungsstufen und Benutzerbeschreibungen anderen Benutzern jederzeit zuweisen, indem sie das AUI Kommando (Assign User ID) eingeben.

5.4.1.1 Format des AUI Kommandos

```
AUI
ASSIGN USER ID
USER DESCRIPTION: (string)
NEW USER ID: <name>
NEW PASSCODE: [ string ]
USER PRIVILEGE CODE (0..7): <int>
```

5.4.1.2 Benutzerantworten beim AUI Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
USER DESCRIPTION:	R	Eine Zeichenkette von 1 bis 16 alphanumerischen Zeichen, um den Benutzer des Terminals zu identifizieren (dient nur zu Informationszwecken).
NEW USER ID:	R	Eine sechs Zeichen lange Benutzeridentifikation im folgenden Format: aaannn dabei sind: aaa drei Buchstaben nnn drei Ziffern, die nur für diesen speziellen Benutzer gelten. Um Platz in der Datei zu sparen, sollte dies die kleinste noch nicht in Gebrauch befindliche Nummer sein.
NEW PASSCODE:	O	Eine Zeichenkette von 1 bis 6 alphabetischen oder numerischen Zeichen, die den Passcode darstellt, der der bei NEW USER ID angegebenen Benutzerkennung zugewiesen wird.
USER PRIVILEGE CODE: (0 . . . 7)	R	Eine ganze Zahl von 1 bis 7, die die Bevorrechtigungsstufe angibt, mit der der Benutzer unter TAXO arbeiten kann.

5.4.1.3 Beispiel zum AUI Kommando

Das folgende Beispiel zeigt die Eingabe eines AUI Kommandos, um Benutzerbeschreibung, Benutzeridentifikation, Passcode und Benutzer-Bevorrechtigungsstufe zuzuweisen.

```
( ) AUI
    ASSIGN USER ID
    USER DESCRIPTION: TECHPUBS
    NEW USER ID: TEC046
    NEW PASSCODE:
    USER PRIVILEGE CODE (0..7): 0
( )
```

5.4.2 Verändern von Benutzeridentifikationen - MUI Kommando

Autorisierte Benutzer können vorhandene Benutzeridentifikationen und Passcodes ändern, indem sie das MUI Kommando eingeben. Wenn ein Benutzer gegenwärtig nach einem Log-in unter einer Benutzeridentifikation arbeitet, die geändert werden soll, so wird diese nicht modifiziert und es wird die folgende Fehlermeldung ausgegeben:

**** ERROR 9021 **** USER ID IN USE

Als Benutzer kann man seine eigene Identifikation ändern, wenn man darauf achtet, nicht noch einen Batch-Lauf im System zu haben.

5.4.2.1 Format des MUI Kommandos

MUI
MODIFY USER ID
USER ID: <name>
NEW PASSCODE: [string]
NEW USER PRIVILEGE CODE: [int]

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
USER ID	R	Eine sechs Zeichen lange Benutzeridentifikation
NEW PASSCODE	O	Eine Kette von 1 bis 8 alphabetischen oder numerischen Zeichen, die den Passcode darstellt, der der bei USER ID angegebenen Benutzeridentifikation zugewiesen werden soll. Standardwert: keine Änderung.
NEW USER PRIVILEGE CODE	O	Eine ganze Zahl zwischen 0 und 7, die die neue Bevorrechtigungsstufe angibt, mit welcher der Benutzer unter TAXO arbeiten kann; Standardwert: keine Änderung.

5.4.2.2 Beispiel zum MUI Kommando

Das folgende Beispiel zeigt die Eingabe eines LUI Kommandos (List User ID), um die vorhandenen Benutzeridentifikationen, Passcodes und Bevorrechtigungsstufen anzuzeigen, die den jeweiligen Benutzern zugewiesen sind.
Ein MUI Kommando wurde danach eingegeben, um Benutzeridentifikation, Passcode und Bevorrechtigungsstufe für einen Benutzer zu ändern; dann folgte ein weiteres LUI Kommando, um den geänderten Status des Benutzers zu zeigen.

```
() LUI
   LIST USER IDS
    OUTPUT ACCESS NAME:    LP01
   ()
```

USER ID	DESCRIPTION	PRIVILEGE CODE
SYS000	SYSTEM	7
TCR001	TOM	7
CLN002	LARRY	7
MSB003	MARY B	7
MFP004	MIKE P	7
RTE005	RAY E	7
DJE006	DORIS	7
MLT007	MARY T	7
KAC009	KEITH C	7
HRC010	BOB C	7
KWD011	KENNETH D	7
GMB012	GEORGE B	7
HCH013	HERB	7
MLK014	MIKE	7
WHT016	WAYNE	7
GARY17	GARY THE K	7
KOC018	KEITH	7
TKG020	TERRY	7
GPS021	GARRY P.	7
MCS024	MATT S	7
DUM025	DUMMY	7
GEB029	BARRY	7
DUM032	DUMMY	7
DUM034	DUMMY	7
SCB035	STEVE B.	7
DBP037	DATA BASE	7
BBB038	BILLIE	7
BYM042	BRAD	7
JKB044	BERR	7
TEC050	TECHPUBS	0

```
MUI
MODIFY USER ID
 USER ID: SYSTEM  TEC050
 NEW PASSCODE:
 NEW USER PRIVILEGE CODE:    7
()
```

```
LUI
LIST USER IDS
 OUTPUT ACCESS NAME:    LP01
```

USER ID	DESCRIPTION	PRIVILEGE CODE
SYS000	SYSTEM	7
TCR001	TOM	7
CLN002	LARRY	7
MSB003	MARY B	7
MFP004	MIKE P	7
RTE005	RAY E	7
DJE006	DORIS	7
MLT007	MARY T	7
KAC009	KEITH C	7
HRC010	BOB C	7
KWD011	KENNETH D	7
GMB012	GEORGE B	7
HCH013	HERB	7
MLK014	MIKE	7
WHT016	WAYNE	7
GARY17	GARY THE K	7
KOC018	KEITH	7
TXG020	TERRY	7
GPS021	GARRY P.	7
MCS024	MATT S	7
DUM025	DUMMY	7
GEH029	BARRY	7
DUM032	DUMMY	7
DUM034	DUMMY	7
SCB035	STEVE B.	7
DBP037	DATA BASE	7
BBB038	BILLIE	7
BYM042	BRAD	7
JKB044	BERR	7
TEC050	TECHPUBS	7

5.4.3 Löschen von Benutzeridentifikationen - DUI Kommando

Autorisierte Benutzer können das DUI Kommando (Delete User ID) eingeben, um sämtliche Informationen zur Identifikation eines bestimmten Benutzers zu löschen, um diesen vom Zugriff auf TAXO auszuschließen. Wenn ein Benutzer nach einem Log-in unter einer Benutzeridentifikation arbeitet, die gelöscht werden soll, so wird diese nicht gelöscht und es wird die folgende Fehlermeldung ausgegeben:

**** ERROR 9021 **** USER ID IN USE

Als Benutzer kann man seine eigene Identifikation löschen, wenn man darauf achtet, nicht noch einen Batch-Lauf im System zu haben.

5.4.3.1 Format des DUI Kommandos

DUI
DELETE USER ID
USER ID: <name>

5.4.3.2 Benutzerantworten beim DUI Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
USER ID:	R	Die Benutzeridentifikation, die aus dem System entfernt werden soll.

5.4.3.3 Beispiel zum DUI Kommando

Das folgende Beispiel zeigt die Eingabe eines LUI Kommandos (List User ID), um die vorhandenen Benutzeridentifikationen, Passcodes und Bevorrechtigungsstufen anzuzeigen, die den jeweiligen Benutzern zugewiesen sind.
Ein DUI Kommando wurde danach eingegeben, um eine bestimmte Benutzeridentifikation zu löschen und danach wieder ein LUI Kommando, um zu zeigen, daß diese Benutzeridentifikation tatsächlich gelöscht wurde.

```
() LUI
LIST USER IDS
 OUTPUT ACCESS NAME:    ME
 USER ID      DESCRIPTION          PRIVILEGE CODE
    SYS001      SYSTEM                   3
    HERB03      HERB                     7
    EXM005      EXAMPLE USER             4
    KDK007      KIRBY KYLE               7
    JCK008      CLARK KOLB               7
    JCK010      CLARK KOLB               7
    KWD011      KENNETH DUNN             7
    DAR012      DARLENE                  6
    SUE020      SUE                      7
    MLH022      MARILYN HARRIS           7
01:55:30 SATURDAY, JUL 29, 1978.
```

```
() DUI
DELETE USER ID
 USER ID:    EXM005
() LUI
LIST USER IDS
 OUTPUT ACCESS NAME:    ME
 USER ID      DESCRIPTION          PRIVILEGE CODE
    SYS001      SYSTEM                   3
    HERB03      HERB                     7
    KDK007      KIRBY KYLE               7
    JCK008      CLARK KOLB               7
    JCK010      CLARK KOLB               7
    KWD011      KENNETH DUNN             7
    DAR012      DARLENE                  6
    SUE020      SUE                      7
    MLH022      MARILYN HARRIS           7
01:56:13 SATURDAY, JUL 29, 1978.
```

5.4.4 Auflisten von Benutzeridentifikationen - LUI Kommando

Man gibt das LUI Kommando (List User ID) ein, um eine Liste von Benutzern angezeigt zu bekommen, die gegenwärtig autorisiert sind, mit dem System TAXO zu arbeiten. Die am Bildschirm angezeigte Liste beinhaltet die jeweiligen Benutzeridentifikationen, Benutzer-Beschreibungen und die zugeordneten Bevorrechtigungsstufen.

5.4.4.1 Format des LUI Kommandos

LUI
LIST USER IDS
OUTPUT ACCESS NAME: [acnm]

5.4.4.2 Benutzerantworten beim LUI Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
OUTPUT ACCESS NAME:	O	Name des Gerätes oder Pfadname der Datei, wohin TAXO die Liste ausgeben soll. Standardwert ist der Terminal Local File des Terminals.

5.4.4.3 Beispiel zum LUI Kommando

Das folgende Beispiel zeigt die Eingabe eines LUI Kommandos, um eine Liste der Benutzeridentifikationen auf dem Terminal des Benutzers auszugeben.

```
( ) LUI
    LIST USER IDS
    OUTPUT ACCESS NAME:


USER ID     DESCRIPTION     PRIVILEGE CODE
   SYS000     SYSTEM              7
   TCR001     TOM                 7
   CLN002     LARRY               7
   MSB003     MARY                7
   MFP004     MIKE                7
   RTE005     RAY                 7
   DJE006     DORIS               7
   MLT007     MARY                7
   KAC009     KEITH               7
   HRC010     BOB                 7
   KWD011     KENNETH             7
   GMB012     GEORGE              7
   HCH013     HERB                7
   MLK014     MIKE                7
   WHT016     WAYNE               7
   GARY17     GARY THE K          7
   KDC018     KEITH               7
   TKG020     TERRY               7
   GPS021     GARRY P.            7
   DUM025     DUMMY               7
   GEB029     BARRY               7
   DUM032     DUMMY               7
   DUM034     DUMMY               7
   SCB035     STEVE B.            7
   DBP037     DATA BASE           7
   BBB038     BILLIE              7
   BYM042     BRAD                7
   JKB044     BERR                7
```

6 INSTALLIEREN, ZUWEISEN VON ERSATZSTELLEN, ANMELDEN, ABMELDEN UND ANZEIGEN DES STATUS VON PLATTEN

6.1 Allgemeines

Die folgenden Kommandos sind verfügbar, um Platten zu initialisieren, Ersatzstellen für defekte Stellen auf Platten zuzuweisen, Platten beim System an- oder abzumelden und den Status von Platten anzuzeigen:

- IDS Initialize Disk Surface - Analyse von Platten auf defekte Stellen
- ABL Assign Bad Locations On Disk - Zuweisen von Ersatzstellen für defekte Stellen
- RBL Reset Bad Location List On Disk - Aufheben der Ersatzstellenzuweisung
- SBL Show Bad Location List from Disk - Anzeigen der Ersatzstellenliste einer Platte
- INV Initialize New Volume - Initialisieren einer Platte
- IV Install Volume - Installieren einer Platte
- UV Unload Volume - Abmelden einer Platte
- SVS Show Volume Status - Anzeigen des Plattenstatus

6.2 Initialisieren neuer Platten (INV Kommando)

Wird nach dem Montieren einer Platte in ein Plattengerät die Betriebsbereitschaft des Gerätes angezeigt, so kann der Benutzer zur Initialisierung einer neuen Platte das INV Kommando (Initialize New Volume) eingeben. Das INV Kommando schreibt auf die ganze Platte physikalische Sektorformatinformationen (dabei wird ein eventuell vorhandener Datenbestand komplett zerstört), es baut ein neues Master Directory (.VCATALOG) für die Platte auf, weist der Platte einen Plattennamen zu und richtet die Platte ein. Nach dem Initialisieren wird die Platte automatisch beim System angemeldet (IV - Install Volume Kommando). Die Anmeldung gilt, bis die Platte beim Betriebssystem abgemeldet wird (unload) und anschließend entnommen wird.

Platteninhalt auf Spur 0 und 1

Zur optimalen Wiederherstellung (Recovery) werden die Sektoren 0 und 1 der Spur 0 (Volume und Bad ADU Information) während des Initialisierens auf die letzten Sektoren der Spur 1 kopiert. Änderungen auf Spur 0, Sektor 0 durch das MVI Kommando (Modify Volume Information) werden auch auf den zweiten bis letzten Sektor der Spur 1 übertragen.
Informationen zur Analyse von Platten auf defekte Stellen und zur Zuweisung von Ersatzspuren sind auf der Spur 0, Sektor 0 gespeichert (Volume Information).

ANMERKUNG

Die Initialisierung einer neuen Platte dauert zwischen zwei und zehn Minuten. Der wirkliche Zeitaufwand ist von der Größe und Geschwindigkeit der zu initialisierenden Platte abhängig. Eine Platte mit 406 Spuren braucht annähernd zwei Minuten, während eine Platte mit 4056 Spuren etwa 10 Minuten braucht.

6.2.1 Format des INV Kommandos

```
INV
INITIALIZE NEW VOLUME
UNIT NAME: <name> @
VOLUME NAME: <name>@
NUMBER OF VCATALOG ENTRIES: <int>
BAD TRACK ACCESS NAME: [ acnm ]@
DEFAULT PHYSICAL RECORD SIZE: [ int ]
HARDWARE INTERLEAVING FACTOR: [ int ]
FORCE CLEARING OF DISK?: <yes/no>
USED AS SYSTEM DISK?: <yes/no>
LISTING ACCESS NAME:
LOADER ACCESS NAME: [ acnm ]
```

Wird nur dann angezeigt, wenn bei der vorangehenden Meldung mit "YES geantwortet wurde.

6.2.2 Antworten des Benutzers beim INV-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antwort des Benutzers
UNIT NAME	R	Der Gerätename einer Platteneinheit, auf welcher die Platte, die initialisiert werden soll, liegt.
VOLUME NAME	R	Ein aus 1-8 Zeichen bestehender Plattenname (das 1. Zeichen muß ein Alphazeichen sein), mit dem die Platte identifiziert werden kann. Mehrere Platten können den gleichen Plattennamen haben, es kann jedoch nur eine zu einem Zeitpunkt im System angemeldet (installiert) sein.
NUMBER OF VCATALOG ENTRIES:	R	Eine Zahl, welche die max. Anzahl der Eintragungen (Directories, Subdirectories und Dateien) in das Master Volume Directory (VCATALOG) angibt. Standardwert ist 342.
BAD TRACK ACCESS NAME:	O	Der Pfadname einer Datei oder ein Gerätename oder ein Synonym dafür, von wo das INV Kommando Informationen über schadhafte Spuren einlesen kann. Standard ist keine Information über fehlerhafte Spuren. Gibt der Benutzer ME ein, dann werden die Informationen über fehlerhafte Spuren wie folgt am Terminal angefordert: ENTER BAD TRACKS IN THE FORMAT: HEAD, CYLINDER; OR HEAD, CYLINDER; HEAD, CYLINDER; etc. TO END LIST, ENTER AN EMPTY LINE. Die Angaben über schadhafte Spuren werden von INV im Format Kopfnr., Zylindernr; oder Kopfnr., Zylindernr; Kopfnr., Zylindernr. erwartet. Die Angabe von schadhaften Spuren wird durch eine leere Zeile angezeigt. Alle eingegebenen Nummern müssen dezimal sein.
DEFAULT PHYSICAL RECORD SIZE:	O	Die physikalische Satzgröße, die für alle nachfolgenden Dateien verwendet wird, die auf der Platte angelegt werden. Der Standardwert, im Normalfall 864, wird zur Laufzeit von SYSGEN bestimmt.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antwort des Benutzers
HARDWARE INTERLEAVING FACTOR	O	Der Überlappungsfaktor wird für Disketten mit Double Density-Aufzeichnung benötigt. Standardwert ist 1 (Lesen und Schreiben auf fortlaufende Sektoren). Bei Platten wird der Überlappungsfaktor ignoriert.
USED AS SYSTEM DISK?:	R	"Yes" wird eingegeben, wenn die Platte als Systemplatte verwendet werden soll, und das bedeutet, daß auf der Platte ein Loader eingerichtet wird. "No" wird eingegeben, wenn die Platte nicht als Systemplatte verwendet werden soll und deshalb keinen Loader braucht. Standard ist "Yes".
FORCE CLEARING OF DISK?:	O	Bei Eingabe von YES wird die gesamte Platte mit binären Nullen gelöscht, anderenfalls werden nur die systemrelevanten Teile neu initialisiert. Standardwert ist "No".
LISTING ACCESS NAME:	O	Es kann ein Geräte- oder Dateiname angegeben werden, wo INV eine Liste von defekten Spuren ausgibt, falls während der Initialisierung solche auftreten. Bei Folge-Initialisierung müssen defekte Spuren nicht erneut angegeben werden. Sie werden aus der Bad ADU List weiter übernommen.
LOADER ACCESS NAME:	O	Der Zugriffsname einer Datei, die als Loader eingerichtet werden soll, muß eingegeben werden. Erfolgt keine Eingabe, so wird der Standard System Loader auf die Platte geschrieben.

6.2.3 Beispiel zum INV-Kommando

Das folgende Beispiel zeigt die Initialisierung einer Platte mit dem Namen VOL1, die auf der Platteneinheit DS02 eingelegt ist. VOL1 kann maximal 100 Dateien enthalten. Die Platte wird als Systemplatte verwendet und die Benutzerdatei mit dem Zugriffsnamen .LOADER wird als Loader auf die Platte geschrieben.

```
( ) INV
INITIALIZE NEW VOLUME
UNIT NAME:     DS02
VOLUME NAME:   VOL1
NUMBER OF VCATALOG ENTRIES: 342    100
BAD TRACK ACCESS NAME:      .BADTRACK
DEFAULT PHYSICAL RECORD SIZE:   864
HARDWARE INTERLEAVING FACTOR:  1
FORCE CLEARING OF DISK?: Y
USED AS SYSTEM DISK?:   YES
LISTING ACCESS NAME: ME
LOADER ACCESS NAME:   .LOADER
```

6.3 Kommandos zum Zuweisen von Ersatzstellen für defekte Stellen auf einer Platte

6.3.1 Allgemeines

Einige der für die TA 1600/30 verfügbaren Plattenlaufwerke unterstützen das Zuweisen von Ersatzstellen für defekte Stellen auf einer Platte. Die Ersatzstellenzuweisung kann - abhängig vom Laufwerk - spur- oder sektorweise geschehen [1]. Folgende Kommandos stehen dem Benutzer zur Verfügung, um auf einer Platte Ersatzstellen zuzuweisen:

- ABL — Assign Bad Locations (Zuweisen von Ersatzstellen)
- RBL — Reset Bad Locations (Zurücksetzen der zugewiesenen Ersatzstellen)
- SBL — Show Bad Locations (Anzeigen der Ersatzstellenliste einer Platte)
- IDS — Initialize Disk surface (Analyse der Plattenoberfläche)

WARNUNG:

Die Ausführung der Kommandos ABL und RBL für eine Platte hat zur Folge, daß der gesamte Inhalt der Platte gelöscht wird. Sollen deshalb auf einer Platte, die bereits Daten enthält, Ersatzstellen zugewiesen werden, so müssen diese vorher gesichert werden. Nach den Zuweisen der Ersatzstellen muß die Platte mit INV neu initialisiert werden. Anschließend können die Daten wieder zurückkopiert werden.

Bei Datenträgern, bei denen vor dem Initialisieren keine Zuweisung von Ersatzspuren mit ABL möglich ist, (z.B.: Floppy Diskette) muß eine Analyse der Plattenoberfläche durchgeführt werden (IDS-Kommando.)
Das INV-Kommando prüft dies und gibt eine evtl. Fehlermeldung an den Benutzer.
Für Folgeinitialisierungen kann das INV-Kommando wie bisher verwendet werden.

6.3.2 Zuweisen von Ersatzstellen (ABL Kommando)

Das ABL (Assign Bad Locations) Kommando erlaubt einem Benutzer auf einer Platte für defekte Stellen ("Bad Locations", "Bad Tracks", Bad Sectors") Ersatzstellen zuzuweisen. Dabei wird eine auf der Platte bereits abgelegte Ersatzstellenliste gelesen, um die vom Benutzer in einer Datei oder im Dialog angegebenen defekten Stellen erweitert und dann auf die Platte zurückgeschrieben. Beim Zurückschreiben wird die gesamte Platte gelöscht und es werden für die angegebenen defekten Stellen - soweit möglich - Ersatzstellen zugewiesen. Anschließend muß die Platte neu initialisiert werden.

WARNUNG:

Durch die Ausführung des ABL Kommandos wird der gesamte Inhalt der Platte gelöscht.

Das ABL Kommando prüft, ob das angesprochene Laufwerk eine Ersatzstellenzuweisung unterstützt und ob die eingelegte Platte nicht in Benutzung ist. Ist dies der Fall, werden über dem vom Benutzer bei BAD LOCATION ACCESS NAME angegebenen Zugriffsnamen die schadhaften Stellen in folgender Form erwartet:

Kopfnummer, Zylindernummer [,Sektornummer] ;

ANMERKUNG:

1) Die z. Zt. verfügbaren Laufwerke unterstützen nur das spurweise Zuweisen von Ersatzspuren.

Die Angabe der Sektornummer ist nur notwendig, wenn das Laufwerk eine Ersatzsektorzuweisung unterstützt. Bei Ersatzspurzuweisung wird die Angabe der Sektornummer, falls vorhanden, ignoriert. Sollen die Angaben im Dialog gemacht werden, so wird als Eingabeaufforderung folgender Text am Terminal ausgegeben:

```
ENTER BAD LOCATIONS IN THE FORMAT:
HEAD, CYLINDER, SECTOR;
                OR
HEAD, CYLINDER, SECTOR; HEAD, CYLINDER, SECTOR; ETC.
THE SECTOR VALUE IS OPTIONAL IF THE DISK SUPPORTS
BAD TRACK ASSIGNING.
TO END LIST, ENTER AN EMPTY LINE.
>
```

Der Cursor steht hinter dem >Zeichen. Der Benutzer kann nun die defekten Stellen eintasten. Jede Zeile ist durch die Return Taste abzuschließen. Die Eingaben werden durch eine leere Zeile (nur Return Taste drücken!) abgeschlossen.

Die vom Benutzer angegebenen schadhaften Stellen werden in die bereits auf der Platte abgelegten Liste übernommen. Nun wird versucht die Ersatzstellenzuweisung durchzuführen.

ANMERKUNG:

Das Zuweisen der Ersatzstellen benötigt bei CMD Fest/Wechselplatten und bei SMD Wechselplatten pro Oberfläche ca. 40 Sekunden, d.h. für eine CMD 16 MB Fest- oder Wechselplatte ca. 40 sec, für eine CMD 48 MB Festplatte 2 min. und für eine SMD 80 MB Wechselplatte 3 1/2 min. Während dieser Zeit ist es nicht möglich, auf das angesprochene oder auf ein über den gleichen Plattenkontroller angeschlossenes Laufwerk zuzugreifen. Treten während der Ersatzstellenzuweisung Fehler auf, kann sich die benötigte Zeit auf Grund von automatischen Wiederholungen verlängern.

Über das Ergebnis wird folgendes Protokoll erstellt.

Beispiel:

```
THE FOLLOWING DISK LOCATIONS ARE MARKED AS BAD
HEAD, CYLINDER, SECTOR, SPARE LOC ASSIGNED
  0,      20,     10, YES
  1,      30,      5, NO
```

Dieses Protokoll enthält alle als defekt markierten Stellen in der Form Kopfnummer, Zylindernummer, Sektornummer; in der Spalte SPARE LOC ASSIGNED steht YES, wenn für die angegebene defekte Stelle eine Ersatzstelle zugewiesen wurde und NO, wenn keine Ersatzstelle zugewiesen wurde. Die mit NO gekennzeichneten defekten Stellen können beim jetzt notwendigen INV (Initialize new Volume) als "Bad Tracks" angegeben werden. Untersützt das Laufwerk nur eine Ersatzspurzuweisung, wird die Spalte SECTOR nicht ins Protokoll geschrieben.

6.3.2.1 Format des ABL Kommando

```
ABL
ASSIGN BAD LOCATIONS ON DISK
UNIT NAME: <name > @
BAD LOCATION ACCESS NAME: [ acnm ] @
LISTING ACCESS NAME: [ acnm ] @
```

6.3.2.2 Benutzerantworten zum ABL Kommando

Systemmeldungen	Antwort erforderlich (R) oder Optional (O)	Benutzerantworten
UNIT NAME	R	Der Name des Plattenlaufwerks, in den die Platte eingelegt ist, auf der Ersatzstellen zugewiesen werden sollen.
BAD LOCATION ACCESS NAME	R	Der Name eines Gerätes oder der Pfadname einer Datei, von wo die Angaben über defekte Stellen gelesen werden sollen.
LISTING ACCESS NAME	O	Der Name eines Gerätes oder der Pfadname einer Datei, wo das Protokoll über die Ersatzstellenzuweisung erstellt werden soll. Keine Angabe: Das Protokoll wird im Terminal Local File des Benutzers erstellt.

6.3.2.3 Beispiel zum ABL Kommando

Auf einer CMD 16 MB Cartridge ist vom Hersteller folgender Aufkleber mit defekten Spuren aufgeklebt.

TRACKS TO FLAG

TRACK	TRACK	TRACK	TRACK
C644 00			
C645 00			

Dies bedeutet, daß die Zylinder 644 und 645 auf Plattenoberfläche (= Kopf) 0 nicht einwandfrei sind. Für diese Spuren sollten deshalb Ersatzspuren zugewiesen werden. Die Platte wird ins Laufwerk DS04 eingelegt. Nun kann das ABL Kommando wie folgt ausgeführt werden. (Benutzereingaben sind unterstrichen, <cr > bedeutet drücken der Return Taste) :

```
( ) ABL
    ASSIGN BAD LOCATIONS ON DISK
    UNIT NAME: DS04 <cr >
    BAD LOCATION ACCESS NAME: ME <cr >
    LISTING ACCESS NAME:

    ENTER BAD LOCATIONS IN THE FORMAT:
    HEAD, CYLINDER, SECTOR;
              OR
    HEAD, CYLINDER, SECTOR; HEAD, CYLINDER, SECTOR; ETC.
    THE SECTOR VALUE IS OPTIONAL IF THE DISK SUPPORTS
    BAD TRACK ASSIGNING.
    TO END LIST, ENTER AN EMPTY LINE.
    > 0, 644; <cr >
    > 0, 645, <cr >
    > <cr >
```

```
THE FOLLOWING DISK LOCATIONS ARE MARKED AS BAD
HEAD,  CYLINDER;  SPACE LOC ASSIGNED
   0,        644;  YES
   0,        645;  YES

( )
```

Damit sind den beiden Spuren Ersatzspuren zugewiesen. Die Platte kann deshalb jetzt wie eine Platte ohne Fehlerstellen ("Error Free") mit INV initialisiert werden.

6.3.3 Zurücksetzen der zugewiesenen Ersatzstellen (RBL Kommando)

Das RBL (Reset Bad Locations) Kommando erlaubt einem Benutzer, Ersatzstellenzuweisung auf einer Platte aufzuheben.

WARNUNG:

Durch die Ausführung des RBL Kommandos wird der gesamte Inhalt der Platte gelöscht. Die Platte muß anschließend neu initialisiert werden.

Das RBL Kommando prüft vor dem Zurücksetzen, ob das angesprochene Laufwerk eine Ersatzstellenzuweisung zuläßt und ob die eingelegte Platte nicht in Benutzung ist.

6.3.3.1 Format des RBL Kommando

```
RBL
RESET BAD LOCATION LIST ON DISK
UNIT NAME:  <name >@
```

6.3.3.2 Benutzerantworten zum RBL Kommando

Systemmeldung	Antwort erforderlich (R) oder Optional (O)	Benutzerantwort
UNIT NAME	R	Der Name des Laufwerks, in dem die Platte liegt, auf der die Ersatzstellenzuweisung aufgehoben werden soll.

6.3.3.3 Beispiel zum RBL Kommando

Beim Zuweisen von Ersatzspuren mit dem ABL Kommando wurde irrtümlicherweise eine falsche Spur angegeben. Der Benutzer möchte deshalb die Ersatzspurzuweisung korrigieren. Dies ist möglich, indem die Ersatzspurzuweisung mit dem RBL Kommando aufgehoben und anschließend mit ABL durchgeführt wird. Im folgenden Beispiel sind Benutzerantworten unterstrichen. <cr > bedeutet drücken der Return Taste:

```
( ) RBL
    RESET BAD LOCATION LIST ON DISK
    UNIT NAME:  DS04 <cr >

( ) ABL
    ASSIGN BAD LOCATIONS ON DISK
    UNIT NAME:  DS04 <cr >
    BAD LOCATION ACCESS NAME:  ME <cr >
    LISTING ACCESS NAME

    ENTER BAD LOCATIONS IN THE FORMAT:
    HEAD, CYLINDER, SECTOR;
              OR
    HEAD, CYLINDER, SECTOR; HEAD, CYLINDER, SECTOR; ETC.
    THE SECTOR VALUE IST OPTIONAL IF THE DISK SUPPORTS
    BAD TRACK ASSIGNING.
    TO END LIST, ENTER AN EMPTY LINE.
    >0, 644; <cr >
    >0, 645; <cr >
    > <cr >

    THE FOLLOWING DISK LOCATIONS ARE MARKED AS BAD
    HEAD,  CYLINDER;  SPARE LOC ASSIGNED
       0,        644;  YES
       0,        645;  YES

( )
```

6.3.4 Anzeigen der Ersatzstellenliste einer Platte (SBL Kommando)

Mit dem SBL (Show Bad Locations) Kommando kann die Liste der defekten Stellen einer Platte und Informationen ob Ersatzstellen dafür zugewiesen sind, angezeigt werden. Die erzeugte Ausgabe hat das gleiche Format wie das Protokoll, das bei Ausführung des ABL Kommandos erzeugt wird (siehe 2.).

6.3.4.1 Format des SBL Kommandos

```
SBL
SHOW BAD LOCATION LIST FROM DISK
UNIT NAME:  <name > @
LISTING ACCESS NAME:  [ acnm ]  @
```

6.3.4.2 Benutzerantworten zum SBL Kommando

Systemmeldungen	Antwort erforderlich (R) oder Optional (O)	Benutzerantworten
UNIT NAME	R	Name des Laufwerks, in dem die Platte eingelegt ist, von der die Ersatzstellenliste angezeigt werden soll.
LISTING ACCESS NAME	O	Name eines Gerätes oder der Pfadname einer Datei, wo die Anzeige ausgegeben werden soll.

6.3.4.3 Beispiel zum SBL Kommando

Das folgende Beispiel zeigt, wie von der im Laufwerk DS04 eingelegten Platte die Liste der Ersatzstellen angezeigt werden kann:

```
( ) SBL
    SHOW BAD LOCATION LIST FROM DISK
    UNIT NAME: DS04 <cr>
    LISTING ACCESS NAME: <cr>

    THE FOLLOWING DISK LOCATIONS ARE MARKED AS BAD
    HEAD,  CYLINDER;  SPARE LOC ASSIGNED
       0;       644;  YES
       0;       645;  YES

( )
```

6.3.5 IDS Kommando

Das IDS - Kommando analysiert die Plattenoberfläche auf defekte Spuren und markiert diese. Das Kommando prüft jedoch nicht das Plattenlaufwerk oder die Plattensteuereinheit.
IDS kann nicht alle defekten Spuren feststellen, deshalb muß der Anwender die vom Hersteller als defekt gekennzeichneten Spuren eingeben.
Falls defekte Spuren beim Analysieren der Plattenoberfläche festgestellt werden, so wird dies auf einer sogenannten "Defektspuren-Liste" auf der Platte vermerkt.

Jede Plattenoberfläche einer Platte muß vor Verwendung initialisiert werden. Nach Initialisierung der Plattenoberfläche kann mittels des INV - Kommandos der Name der angesprochenen Platte initialisiert werden.
Alternativ erlaubt das IDS Kommando sowohl Initialisierung von Plattenoberfläche als auch des Plattennamens.

6.3.5.1 Format des IDS Kommandos

IDS
INITIALIZE DISK SURFACE
UNIT NAME: <name>@
INITIALIZE NEW VOLUME?: <yes/no>
BAD TRACK ACCESS NAME: [acnm]
LISTING ACCESS NAME: [acnm]

Bei Eingabe von YES als Antwort auf INITIALIZE NEW VOLUME?: wird der folgende Hinweis gezeigt:

VOLUME NAME:	<name>@
NUMBER OF VCATALOG ENTRIES:	<int>
DEFAULT PHYSICAL RECORD SIZE:	[int]
HARDWARE INTERLEAVING FACTOR:	[int]
USED AS SYSTEM DISK?:	[yes/no]
LOADER ACCESS NAME:	[acnm]

6.3.5.2 Benutzerantworten zum IDS Kommando

Systemmeldungen	Antwort erforderlich (R) oder Optional (O)	Benutzerantworten
UNIT NAME:	R	Der Gerätename einer Platteneinheit, auf welcher die Platte, die initialisiert werden soll, liegt.
INITIALIZE NEW VOLUME?:	R	YES bedeutet: Der Plattenname soll initialisiert werden. NO bedeutet: Nur die Plattenoberfläche soll initialisiert werden. Für weitere Informationen siehe INV-Kommando
BAD TRACK ACCESS NAME:	O	Der Pfadname einer Datei oder ein Gerätename, von denen das IDS Kommando Informationen über schadhafte Spuren einlesen kann. Standard ist keine Information über fehlerhafte Spuren. Gibt der Benutzer ME ein, dann werden die Informationen über fehlerhafte Spuren wie folgt am Terminal angefordert: ENTER BAD TRACKS IN THE FORMAT: HEAD, CYLINDER; HEAD, HEAD, CYLINDER; etc. TO END LIST, ENTER AN EMPTY LINE. Die Antwort über defekte Spuren muß im gewünschten Format eingegeben werden. Alle Zahlen müssen in dezimaler Form angegeben werden.
LISTING ACCESS NAME:	O	Der Pfadname einer Datei oder eines Gerätes, wo das Protokoll über das Ergebnis der Platten-Oberflächeninitialisierung erstellt werden soll. Keine Angabe: Das Protokoll wird im Terminal local file des Benutzers erstellt.

Das INV Kommando gibt Aufschluß über Antworten auf folgende Benutzer Hinweise:

VOLUME NAME:	<name>@
NUMBER OF VCATALOG ENTRIES:	<int>
DEFAULT PHYSICAL RECORD SIZE:	[int]
HARDWARE INTERLEAVING FACTOR:	[int]
USED AS SYSTEM DISK?:	[yes/no]
LOADER ACCESS NAME:	[acnm]

6.3.5.3 Beispiel zum IDS Kommando

Im folgenden Beispiel initialisiert das IDS Kommando die Plattenoberfläche im Gerät mit dem Namen: DS02
Die Platte darf nicht schreibgeschützt und noch nicht installiert sein.

```
( ) IDS
  INITIALIZE DISK SURFACE
  UNIT NAME:                    DS02
  INITIALIZE NEW VOLUME?:       NO
  BAD TRACK ACCESS NAME:        .BADTRACK
  LISTING ACCESS NAME:
```

6.4 Installieren von Platten (IV-Kommando)

Der Benutzer muß eine initialisierte Platte in eine Platteneinheit einlegen, die Betriebsbereitschaft der Platteneinheit abwarten und kann dann ein IV-Kommando (Install Volume) eingeben, um eine Platte für weitere Operationen zu installieren. TAXO kann ein Volume nicht installieren, wenn der mit dem IV-Kommando spezifizierte Name der Platte nicht mit dem Plattennamen übereinstimmt, der der Platte, die installiert werden soll, zugewiesen wurde.

ANMERKUNG

Ist ein Plattenantrieb physikalisch in eine Wechsel- und eine Festplatte unterteilt, der durch eine Plattenscheibenspindel angetrieben wird, so ist ein Zugriff auf die Festplatte nicht möglich, wenn sich die Wechselplatte in der Lade- oder Entladephase befindet.

Das Austauschen der Wechselplatte erfordert das Entladen (UV-Kommando) der Platte und den Abbruch aller Systemaktivitäten. Der Benutzer wechselt die Wechselplatte und richtet die zweite Platte (IV-Kommando) ein.
Des weiteren darf die Platte, die installiert wird, nicht den gleichen Plattennamen einer bereits eingerichteten Platte, einschließlich der Systemplatte haben.

6.4.1 Format des IV-Kommandos

IV
INSTALL VOLUME
UNIT NAME: <name> @
VOLUME NAME: <name> @

6.4.2 Antworten des Benutzers beim IV-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
UNIT NAME:	R	Der Gerätename einer Platteneinheit, auf der die Platte installiert werden soll. Die Platte muß in dem angegebenen Gerät sein.
VOLUME NAME:	R	Eine aus 1-8 Zeichen bestehende Zeichenkette, welche die Platte, die installiert werden soll, bezeichnet.

6.4.3 Beispiel zum IV-Kommando

Das folgende Beispiel zeigt, wie eine Platte mit dem Plattennamen VOL1 auf der Platteneinheit DS02 installiert wird.

```
( ) IV
    INSTALL VOLUME
    UNIT NAME: DS02
    VOLUME NAME: VOL1
( )
```

6.5 Abmelden von Platten (UV-Kommando)

Installierte Platten müssen durch die Eingabe eines UV-Kommandos (Unload Volume) beim Betriebssystem abgemeldet werden, bevor sie aus der Platteneinheit entnommen werden können. Benutzer, die eine Platte im installierten Zustand ohne Eingabe eines UV-Kommandos entnehmen, verhindern damit, daß andere Benutzer die verwendete Platteneinheit benutzen können (siehe SVS-Kommando). Nachdem die Platte abgemeldet ist, zeigt TAXO in einer Meldung an, welche Platteneinheit abgemeldet wurde.

ANMERKUNG

Ist ein Plattenantrieb physikalisch in eine Wechsel- und eine Festplatte unterteilt, der durch eine Plattenscheibenspindel angetrieben wird, so ist ein Zugriff auf die Festplatte nicht möglich, wenn sich die Wechselplatte in der Lade- oder Entladephase befindet.

Das Austauschen der Wechselplatte erfordert das Entladen (UV-Kommando) der Platte und den Abbruch aller Systemaktivitäten. Der Benutzer wechselt die Wechselplatte und richtet die zweite Platte (IV-Kommando) ein.
Des weiteren darf die Platte, die installiert wird, nicht den gleichen Plattennamen einer bereits eingerichteten Platte, einschließlich der Systemplatte haben.

6.5.1 Format des UV-Kommandos

```
UV
UNLOAD VOLUME
VOLUME NAME: <name> @
```

6.5.2 Antworten des Benutzers beim UV-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
VOLUME NAME	R	Der Name der installierten Platte, die abgemeldet werden soll.

6.5.3 Beispiel zum UV Kommando

Das folgende Beispiel zeigt die Eingabe des UV Kommandos, um die installierte Platte mit dem Namen VOL1 abzumelden.

```
( ) UV
    UNLOAD VOLUME
    VOLUME NAME: VOL1
( )
```

6.6 Anzeigen des Plattenstatus (SVS Kommando)

Benutzer können ein SVS Kommando (Show Volume Status) eingeben, um den Status einer Platte drucken oder anzeigen zu lassen. Tabelle 6-1 beschreibt die Information, die durch ein SVS Kommando angezeigt oder gelistet wird.

Tabelle 6-1 Durch ein SVS Kommando erzeugte Information

Feld	Beschreibung
VOLUME NAME:	Der Name der angesprochenen Platte eines Plattenantriebes.
ADUS:	Die Summe des auf der Platte für Directories, Subdirectories und Dateien verfügbaren Speicherplatzes (in ADUs).
#BAD:	Die Anzahl schlechter (fehlerhafter) ADUs auf der Platte.
BYTES/ADU:	Die Anzahl der Bytes pro ADU auf der Platte
AVAILABLE:	Die Summe der ADUs, die noch für Directories, Subdirectories und Dateien verfügbar ist.
LARGEST AVAILABLE BLOCK:	Die in ADUs angegebene Größe des größten zusammenhängenden Blockes, der für eine Datei verfügbar ist.
CONTROLLER ERRORS:	Die Anzahl erfolgloser Versuche seit der Betriebssysteminitialisierung, um Daten auf die Platte zurückzuschreiben oder von dort wieder zu lesen, wenn durch den Plattencontroller ein Fehler erkannt wurde. Dieser Wert sollte normalerweise Null sein. Jedoch sollte ein Wert zwischen 1 und 10 während einer Woche kein Anlaß zur Beunruhigung sein.
PRIMARY SYSTEM IMAGE:	Wenn die Platte eine Systemplatte ist, wird der Name des Primary System Image angegeben.
SECONDARY SYSTEM IMAGE:	Wenn die Platte eine Systemplatte ist, wird der Name des Secondary System Image angegeben.

Feld	Beschreibung
NAME INSTALLED:	Der Name der Platte, welche gerade installiert ist. Dieses Feld wird mit Blanks ausgefüllt, wenn keine Platte installiert ist. Wenn der in diesem Feld angegebene Name nicht der gleiche wie der vorher angegebene Plattenname ist, so wurde die Platte aus dem Antrieb genommen, ohne daß ein UV-Kommando angegeben wurde. Ein UV-Kommando muß für diesen Namen eingegeben werden, bevor eine andere Platte installiert werden kann. Wenn die Platte von einem DCOPY oder INV benutzt wird, ist dieses Feld mit "$$$$$$$$" Zeichen gefüllt und kann nicht entladen werden.

6.6.1 Format des SVS-Kommandos

SVS
SHOW VOLUME STATUS
VOLUME NAME: <name> @
DRIVE NAME: <name> @
OUTPUT ACCESS NAME: <acnm> @

6.6.2 Antworten des Benutzers beim SVS-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Antworten des Benutzers
VOLUME NAME:	O	Der Name der Platte, dessen Status angezeigt werden soll. Wird diese Antwort weggelassen, so muß der Drive Name (Plattenantriebsname) angegeben werden. Werden sowohl Drive Name, wie auch der Volume Name angegeben, so wird der Volume Name verwendet.
DRIVE NAME	O	Der Gerätename der Platteneinheit, auf welcher die Platte eingerichtet (installiert) ist. Fehlt der Drive Name, so muß der Volume Name angegeben werden. Werden beide Namen angegeben, so wird der Volume Name verwendet.
OUTPUT ACCESS NAME:	O	Der Gerätename oder der Datei-Pfaname, auf die TAXO die durch das SVS-Kommando erzeugte Statusinformation schreiben soll. Standard ist das Terminal Local File des aufrufenden Terminals.

6.6.3 Beispiel zum SVS-Kommando

Das folgende Beispiel zeigt die Eingabe eines SVS-Kommandos, um den Status einer auf der Platteneinheit DS02 eingelegten Platte auf dem Terminal des Benutzers anzeigen zu lassen.

```
() SVS
SHOW VOLUME STATUS
 VOLUME NAME:
 DRIVE NAME:  DS02
 OUTPUT ACCESS NAME:    ME

VOLUME NAME:  HAWK2    ADUS:  16320   # BAD:  0   BYTES/ADU:  288
AVAILABLE: 4519  LARGEST AVAILABLE BLOCK:  4519   CONTROLLER ERRORS:
PRIMARY SYSTEM IMAGE:  CC4       SECONDARY SYSTEM IMAGE:
NAME INSTALLED:  HAWK2
()
```

7 ANLEGEN UND LÖSCHEN VON DATEIEN UND DIRECTORIES

7.1 Allgemeines

TAXO unterstützt 6 Dateiarten wie sie in Tabelle 7-1 beschrieben sind.

Folgende Kommandos stehen für das Anlegen von Dateien - deren Arten in Tabelle 7-1 beschrieben sind - zur Verfügung:

- CF (Create File)
- CFREL (Create RELative record File)
- CFSEQ (Create SEQuential File)
- CFKEY (Create KEY Indexed File)
- CFDIR (Create DIRectory File)
- CFPRO (Create PROgram File)
- CFIMG (Create IMaGe File)
- DD (Delete Directory)
- DF (Delete File)

Tabelle 7-1 Von TAXO unterstützte Dateitypen

Dateityp	Beschreibung
Relative Record	Relativ Record Dateien haben feste Satzlängen, der Zugriff erfolgt wahlfrei über Satznummern. Ein physikalischer Satz (Einheiten für den Plattenzugriff) kann aus einigen logischen Sätzen bestehen.
Sequentiell	Sequentielle Dateien haben variable Satzlängen, auf die Sätze kann nur sequentiell zugegriffen werden. Die Länge der Sätze kann gerade, ungerade oder Null sein.
Key Indexed	Key Indexed-Dateien haben variable Satzlängen, auf ihre Sätze kann direkt oder sequentiell über den Wert eines jeden der 14 verschiedenen Schlüssel zugegriffen werden. Die Länge der Sätze darf weder ungerade noch Null sein.
Programm	Programmdateien sind Relativ Record-Dateien, die lauffähige Programme - bestehend aus Tasks, Procedures und Overlays - enthalten.

Directory	Directory Dateien sind Relativ Record-Dateien, die Zeiger auf Dateien und andere Directories beinhalten.
Image	Image Files sind Relativ Record-Dateien, deren Satzlänge von der physikalischen Aufteilung der Platte abhängt. Es sind dies Dateien für ganz bestimmte Zwecke, meist von TAXO für Speicherabbilder, roll-in/roll-out und Dumps für Prüfzwecke benutzt.

7.2 Anlegen einer beliebigen Datei (CF-Kommando)

Mit dem CF (Create File) Kommando kann jede Datei, gleich welchen Typs, angelegt werden.

7.2.1 Format des CF Kommandos

CF
CREATE FILE - SEQ, REL, KEY, DIR, PRO, IMG
FILE TYPE:

7.2.2 Benutzerantworten beim CF Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
FILE TYPE	R	Eine der folgenden Abkürzungen:

Abkürzung	Dateityp
DIR	DIRectory Datei
IMG	IMaGe Datei
KEY	KEY Indexed Datei
PRO	PROgram Datei
REL	RELative Record Datei
SEQ	SEQuentielle Datei

Die nachfolgenden Systemmeldungen unterscheiden sich je nach Typ der angegebenen Datei. Die Eingabe des CF-Kommandos mit Angabe des Dateityps ist gleichbedeutend mit der Eingabe des speziellen Create File-Kommandos für den entsprechenden Dateityp.

7.2.3 Beispiele zum CF-Kommando

Das folgende Beispiel zeigt, wie mittels des CF-Kommandos eine Relativ Record Datei angelegt werden kann. Beachten Sie die Ähnlichkeiten in den Systemmeldungen, welche nach diesem CF-Kommando bzw. beim Anlegen einer Datei mittels des CFREL-Kommandos erscheinen.

```
() CF
CREATE FILE - SEQ, REL, KEY, DIR, PROG, IMG
 FILE TYPE: SEQ  REL
CREATE RELATIVE RECORD FILE
 PATHNAME:    VOL1.FILE2
 LOGICAL RECORD LENGTH:    80
 PHYSICAL RECORD LENGTH:    >230
 INITIAL ALLOCATION:    500
 SECONDARY ALLOCATION:    1000
 EXPANDABLE?: YES
 FORCED WRITE?: NO  YES
()
```

7.3 Anlegen von Directory-Dateien (CFDIR-Kommando)

Das Kommando CFDIR (Create File Directory) legt spezielle Dateien, genannt Directories, an, welche Zeiger auf andere Dateien und Subdirectories, die unter dem Directory verzeichnet sind, enthalten. Jedes Directory und Subdirectory einer Hierarchie muß durch ein eigenes CFDIR-Kommando angelegt werden. So kann beispielsweise eine Datei durch folgenden Pfadnamen gekennzeichnet sein: VOL1.SUBDIR.FILE1. Hierbei wurden sowohl das Directory als auch das Subdirectory in dieser Hierarchie durch getrennte Eingabe von CFDIR-Kommandos erzeugt, die folgende Directories anlegten:

VOL1.DIR und VOL1.DIR.SUBDIR.

Die Dateien in Directories oder Subdirectories werden getrennt davon durch eigens dafür vorhandene Kommandos angelegt.

7.3.1 Format des Kommandos CFDIR

CFDIR
CREATE DIRECTORY FILE
PATHNAME: <acnm> @
MAX ENTRIES: <int>
DEFAULT PHYSICAL RECORD SIZE: <int>

7.3.2 Benutzerantworten zum CFDIR-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Der Pfadname des Directories. Alle in dieser Hierarchie darüberliegenden Directories müssen im Pfadnamen enthalten sein.
MAX ENTRIES:	R	Eine Ganzzahl, die die maximale Anzahl der Namen der Dateien angibt, die dieses Directory aufnehmen soll. Die eingegebene Zahl wird auf die nächstliegende Primzahl aufgerundet. Die maximale Anzahl an Dateien eines Directories sollte 512 nicht übersteigen.
DEFAULT PHYSICAL RECORD SIZE:	O	Eine gerade Ganzzahl, welche die physikalische Satzlänge angibt, die als Standardwert beim Anlegen von Dateien innerhalb dieses Directories verwendet werden soll. Wird nichts angegeben, gilt der Wert, der beim Einrichten der neuen Platte angeführt wurde. (Siehe Paragraph 6.2.2)

7.3.3 Beispiel zum CFDIR-Kommando

Bestehen bereits Directories mit den Pfadnamen VOL1.SOURCE und VOL1.SOURCE.MYFILE, so kann man ein Directory mit dem Pfadnamen VOL1.SOURCE.MYFILE.TEST anlegen, indem man das Kommando CFDIR wie folgt benutzt:

```
( ) CFDIR
    CREATE DIRECTORY FILE
    PATHNAME: .SOURCE.MYFILE.TEST
    MAX ENTRIES: 100
    DEFAULT PHYSICAL RECORD SIZE:
( )
```

CFDIR legt ein Directory mit dem angegebenen Pfadnamen an, das 101 (nächste Primzahl) Einträge zuläßt.

ANMERKUNG

Ist im Pfadnamen keine Platte angegeben, verwendet TAXO die Systemplatte.

7.4 Anlegen von Key Index Dateien (CFKEY-Kommando)

Um eine Key Index-Datei zu erhalten, gibt man das CFKEY-Kommando ein. Die genaue Spezifikation und Größe einer Key Index-Datei hängt sehr stark vom vorgesehenen Gebrauch der Datei ab.

7.4.1 Format des CFKEY-Kommandos

CFKEY
CREATE KEY INDEXED FILE
PATHNAME: acnm
LOGICAL RECORD LENGTH: <int>
PHYSICAL RECORD LENGTH: [int]
INITIAL ALLOCATION: [int]
SECONDARY ALLOCATION: [int]
MAXIMUM SIZE: <int>
START POSITION: <int>
KEY LENGTH: <int>
DUPLICATES?: <yes/no>
MODIFIABLE?: <yes/no>
ANY MORE KEYS?: <yes/no>

Diese Einträge können wiederholt werden

7.4.2 Benutzerantworten zum CFKEY-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Der Pfadname der Key Index-Datei
LOGICAL RECORD LENGTH	R	Die charakteristische Größe der logischen Sätze in dieser Datei. Es muß eine gerade Anzahl von Bytes sein. Diese Meldung bestimmt die Länge, auf die die Sätze expandiert werden und gibt die logische Satzlänge an, die auf einen Read File Characteristic SVC zurückgegeben wird. Sie wird nicht benutzt, um den Speicherbedarf zu berechnen.
PHYSICAL RECORD LENGTH	O	Eine gerade Ganzzahl, welche die physikalische Satzlänge angibt. Die physikalische Satzlänge sollte wenigstens dreimal so groß wie die logische Satzlänge sein.
INITIAL ALLOCATION	O	Eine gerade Ganzzahl, welche die Größe des Speicherbedarfs der anzulegenden Datei auf der Platte in logischen Sätzen angibt. Anmerkung: Dieser Parameter wird in Verbindung mit dem "Maximum Size" - Parameter gebraucht, um die Größe des Speicherplatzes zu berechnen, der durch das CFKEY-Kommando angelegt wird. Grob geschätzt, sollte die INITIAL ALLOCATION um mindestens 25 % größer sein als die MAXIMUM SIZE.

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
SECONDARY ALLOCATION	O	Eine gerade ganze Zahl, welche den Speicherplatzbedarf (in logischen Sätzen) angibt, falls die Datei größer wird, als ursprünglich angegeben. Wird keine Secondary Allocation angeführt, gilt eine Standardvorgabe. Key Indexed Files sind immer erweiterbar.
MAXIMUM SIZE	R	Geschätzte maximale Anzahl der erwarteten logischen Sätze. Dieser Parameter wird gebraucht, um Speicherplatz für den Datenteil der Datei anzulegen.
START POSITION	R	Eine ganze Zahl, welche den Spaltenindex des ersten Zeichens im Key angibt. Ein Key, der am Anfang eines Satzes steht, hat die Startposition 1.
KEY LENGTH:	R	Eine ganze Zahl, welche die Zahl der Zeichen im Key angibt (maximal 100). Da die Keywerte in eigenen Verwaltungssätzen gespeichert werden, vergrößern lange Keywerte den Verwaltungsbereich der Datei.
DUPLICATES?	R	Y (yes) wird angegeben, wenn die Datei Sätze mit gleichen Keywerten enthalten kann; N (no) wenn dies nicht der Fall sein kann.
MODIFIABLE?:	R	Y (yes) wird eingegeben, wenn die Werte im Key geändert werden können; N (no), wenn nicht. Der erste Key (Key Nummer 1) kann nicht geändert werden.
ANY MORE KEYS?:	O	"YES" wird eingegeben, wenn der Benutzer weitere Schlüssel definieren will. N (no) zeigt an, daß keine weiteren Schlüssel definiert werden sollen. Ist N (no) interaktiv angegeben, legt das System die Datei an. Wird nichts eingegeben, so wird N (no) als Eingabe angenommen.

7.4.3 Beispiel zum CFKEY-Kommando

Das folgende Beispiel zeigt, wie man interaktiv eine Key Index-Datei mit drei Schlüsseln anlegt.

```
() CFKEY

CREATE KEY INDEXED FILE
 PATHNAME:    VOL1.FILE1

 LOGICAL RECORD LENGTH:    80

 PHYSICAL RECORD LENGTH:    560

 INITIAL ALLOCATION:    1200

 SECONDARY ALLOCATION:    500

 MAXIMUM SIZE:    1000

 EXPANDABLE?: YES

KEY DESCRIPTION
 START POSITION: 0  1

 KEY LENGTH:    2

 DUPLICATES?: NO

 MODIFIABLE?: NO

 ANY MORE KEYS?: YES

KEY DESCRIPTION
 START POSITION: 1  3

 KEY LENGTH:    4

 DUPLICATES?: NO

 MODIFIABLE?: NO  YES

 ANY MORE KEYS?: YES

KEY DESCRIPTION
 START POSITION: 3  7

 KEY LENGTH:    8

 DUPLICATES?: NO  YES

 MODIFIABLE?: NO

 ANY MORE KEYS?: YES  NO
 ()
```

7.4.4 CFKEY im BATCH MODUS

Um eine Key Index-Datei im Batch Modus anzulegen, muß der Batch stream ein CFKEY-Kommando, gefolgt von 1 bis 14 KEY-Kommandos und einem ENDKEY-Kommando, enthalten.

Beispiel:	Kommando		Kommentar
	CFKEY	Parameter	legen eine Key Index-Datei an
	KEY	Parameter	gibt zwischen einem und 14 Schlüssel an
	.	.	
	.	.	
	.	.	
	KEY	Parameter	
	ENDKEY		beendet den Batch-Modus CFKEY

Die Batch Parameter für CFKEY enthalten PATHNAME, LOGICAL RECORD LENGTH (= logische Satzlänge), PHYSICAL RECORD LENGTH (= physikalische Satzlänge), INITIAL ALLOCATION, SECONDARY ALLOCATION und MAXIMUM SIZE. Die Batch-Parameter für jedes KEY-Kommando sind START POSITION, KEY LENGTH, DUPLICATES'?, und MODIFIABLE?. Das ENDKEY-Kommando kennt keine Parameter. Die gezeigte Reihenfolge dieser Kommandos kann, ohne Fehler zu erzeugen, nicht geändert werden. Die Parametereingaben müssen mit den früher bei CFKEY-Parametern gemachten Angaben im interaktiven Modus übereinstimmen.

7.4.5 Neue Key Indexed File Organisation

In TAXO V 1.3.0. wurden in Key Indexed Files die Sätze nach einem Hash Algorithmus abgelegt. In TAXO V 2.0.1 ist dieses Verfahren durch eine Methode, genannt "Sequential Placement", ersetzt. Diese neue Methode ist nicht kompatibel zu dem bisherigen Hash Verfahren. Dies hat zur Folge, daß alle Key Index Files, die unter TAXO V 1.3.0. erstellt wurden und unter TAXO V 2.0.1 weiterverwendet werden sollen, auf das neue Verfahren umgestellt werden müssen. Die prinzipielle Vorgehensweise zum Umkopieren von "hashed" Key Index Files in "sequential" Key Indexed Files wird später noch beschrieben.

7.4.5.1 Formeln zur Berechnung der Größe von Key Indexed Dateien

Mit Einführung des "Sequential Placement" Algorithmus für indexsequentielle Dateien ist es möglich, die maximale Größe einer Datei genau zu berechnen. Die Genauigkeit des Endwertes hängt wesentlich davon ab, wie genau die Werte sind, die der Bediener beim Anlegen einer Datei angibt.

Die Parameter, die zur Berechnung des Plattenbedarfs einer Indexsequentiellen Datei notwendig sind, heißen:

physikalische Satzlänge	(PRS)
durchschnittliche Länge eines logischen Satzes mit "Blank Compression"	(LRS)
Länge aller Schlüssel	(KS)
ADU Größe der Platte, auf welcher die Datei angelegt werden soll	(ADU)
maximale Anzahl an logischen Sätzen	(#LR)
ist die Eingabe der Sätze sortiert	(SPLIT)

Der schwierigste Parameter ist LRS, da hier die durchschnittliche Länge eines Satzes mit "Blank Compression" zu bestimmen ist. LRS ist die durchschnittliche Länge eines logischen Satzes ohne Leerstellen. Die Genauigkeit dieses Parameters bestimmt wesentlich die Qualität des Ergebnisses.

Die Plattenbelegung einer Key Indexed Datei kann in drei Bereiche unterteilt werden:

1. Sicherungsbereich für Update-Vorgänge (Prelogging Area)
2. Bereich der B-Baumknoten (B-Tree Node Area)
3. Bereich der Datenblöcke (Data Block Area)

Anmerkung:

In den folgenden Formeln bedeutet:

glb (v): größte ganze Zahl kleiner oder gleich v (greatest lower bound)

lub (w): kleinste Zahl größer oder gleich w (lowest upper bound)

$\sum_{i=1}^{k}$ (vi): Summe der Werte v1, v2, ..., vk

7.4.5.2 Prelogging Area

Die Prelogging Area ist der einzige Bereich, der eine feste Länge besitzt. Es gilt für die Größe der Prelogging Area:

$$NPRprelog = (18 * K) + 3$$

mit K = Anzahl der Schlüssel

7.4.5.3 B-Tree Node Area

Für die Berechnung dieses Bereiches gelten folgende Formeln:

$$X = glb \left(\frac{PRS - 20}{KS + 6}\right)$$

$$NPRb\text{-}tree = lub \left(\frac{\#LR}{X}\right) + lub \left(SPLIT * lub \left(\frac{\#LR}{X}\right)\right)$$

mit: PRS = physikalische Satzlänge

#LR = maximale Anzahl der logischen Sätze

KS = Länge des Schlüssels

SPLIT = 0.1 falls die Eingabe bezüglich dieses Schlüssels sortiert ist, sonst 0.25

Die Berechnung muß für jeden Schlüssel in der Datei durchgeführt werden. Sind z.B. in der Datei drei Schlüssel vorhanden, müssen drei B-tree Werte für die endgültige NPRb-tree Summierung berechnet werden.

7.4.5.4 Data Block Area

Für die Berechnung dieses Bereiches gelten folgende Formeln:

$$X = \text{glb}\left(\frac{PRS - 16}{LRS + 6}\right)$$

$$NPRdata = \text{lub}\left(\frac{\#LR}{X}\right)$$

mit: PRS = physikalische Satzlänge

#LR = maximale Anzahl der logischen Sätze

LRS = durchschnittliche Länge eines logischen Satzes mit "blank compression" (falls die Datei nur einen Schlüssel besitzt, sollte dessen Länge nicht bei der Bestimmung von LRS mitgerechnet werden, d.h. es ist so zu tun, als bestünde der Schlüssel nur aus Leerstellen)

Die 16 Bytes die von PRS abgezogen werden, setzen sich wie folgt zusammen:

14 Bytes Steuerinformationen am Anfang jedes Satzes

2 Bytes am Ende jedes Satzes.

7.4.5.5 Gesamtgröße

Die Gesamtzahl der benötigten physikalischen Sätze wird nun wie folgt berechnet:

$$NPRtotal = NPRprelog + NPRdata + \sum_{i=1}^{K} (NPRb\text{-}tree\ (i))$$

mit K = Anzahl der Schlüssel

Damit läßt sich die Gesamtzahl der ADU's, die für eine Key Indexed Datei benötigt wird, wie folgt bestimmen:

if PRS > = ADU

then $\#ADU := \text{lub}\left(\frac{PRS}{ADU}\right) * NPRtotal$

else $\#ADU := \text{lub}\left(\frac{1}{\text{glb}\left(\frac{ADU}{PRS}\right)} * NPRtotal\right)$

mit PRS = physikalische Satzlänge

ADU = ADU Größe der Platte

NPRtotal = Gesamtzahl der benötigten physikalischen Sätze

#ADU = Gesamtzahl der benötigten ADU's auf der betreffenden Platte.

7.4.~.6 <u>Beispiel 1</u>

PRS = 864
ADU = 864
LRS = 60
KS = 20
K = 1
#LR = 800
SPLIT = 0.1, da sortierte Eingabe

(A) NPRprelog = (18 * 1) + 3 = 21

(B) $$\text{glb}\left(\frac{864 - 20}{20 + 6}\right) = 32$$

$$\text{lub}\left(\frac{800}{32}\right) + \text{lub}\left(0.1 * \text{lub}\left(\frac{800}{32}\right)\right) = 25 + 3$$

NPRb-tree = 28

(C) $$\text{glb}\left(\frac{864 + 16}{60 + 6}\right) = 12$$

$$\text{NPRdata} = \text{lub}\left(\frac{800}{12}\right) = 67$$

NPRtotal = NPRprelog + NPR-tree + NPRdata

= 21 + 28 + 67

= 116

$$\#\text{ADU} = \text{lub}\left(\frac{864}{864}\right) * 116 = 116$$

7.4.5.7 Beispiel 2

PRS = 864
ADU = 864
LRS = 60
KS1 = 20
KS2 = 20
KS3 = 20
K = 3
#LR = 2600
SPLIT = 0.25, da sortierte Eingabe

(A) NPRprelog = (18 * 3) + 3 = 57

(B) $$glb\ \left(\frac{864 - 20}{20 + 6}\right) = 32$$

$$lub\ \left(\frac{2600}{32}\right) + lub\ \left(0.25 * lub\ \left(\frac{2600}{32}\right)\right) = 82 + 21$$

NPRb-tree (1) = 103 Schlüssel 1
NPRb-tree (2) = 103 Schlüssel 2
NPRb-tree (3) = 103 Schlüssel 3

(C) $$glb\ \left(\frac{864 - 16}{60 + 6}\right) = 12$$

$$NPRdata = lub\ \left(\frac{2600}{12}\right) = 217$$

NPRtotal = NPRprelog + NPRb-tree (1) + NPRb-tree (2) + NPRb-tree (3) + NPRdata

= 57 + 103 + 103 + 103 + 217

= 583

#ADU $$= lub\ \left(\frac{864}{864}\right) * 583 = 583$$

7.5 Anlegen einer Relativ Record Datei (CFREL-Kommando)

Um eine Relativ Record-Datei anzulegen, gibt man das CFREL-Kommando ein.

7.5.1 Format des CFREL-Kommandos

```
CFREL
CREATE RELATIVE RECORD
PATHNAME: <acnm> @
LOGICAL RECORD LENGTH: [ int]
PHYSICAL RECORD LENGTH: [ int]
INITIAL ALLOCATION: [ int]
SECONDARY ALLOCATION: [ int]
EXPANDABLE?: <yes/no>
FORCED WRITE?: <yes/no>
```

7.5.2 Benutzerantworten beim CFREL-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME	R	Der Pfadname der Relativ Record-Datei.
LOGICAL RECORD LENGTH	O	Die Satzlänge in dieser Datei. Standardwert ist 80.
PHYSICAL RECORD LENGTH	O	Die Größe der Plattenübertragungseinheit. Der Standardwert wird vom Benutzer bei der Generierung des Systems festgelegt.
INITIAL ALLOCATION	O	Die Anfangsgröße der Datei (in logischen Sätzen). Das System nimmt einen Standardwert, falls nichts angegeben wird.
SECONDARY ALLOCATION	O	Falls die Datei größer wird als in der Anfangsgröße angegeben, gibt diese Anzahl logischer Sätze die Größe des Wachstumssegments an. Wird kein Wert angegeben, gilt ein Standardwert.
EXPANDABLE ?:	R	Y (yes) erlaubt der Datei größer zu werden, als in der INITIAL ALLOCATION angegeben.
FORCED WRITE?:	R	Auf Relativ Record-Plattendateien wird gewöhnlich erst geschrieben, wenn Speicherplatz benötigt wird. Eingabe von Y hat zur Folge, daß logische Sätze sofort geschrieben werden. N (no) läßt verzögertes Schreiben durch TAXO zu.

7.5.3 Beispiel zum CFREL Kommando

Das folgende Beispiel gibt an, wie man durch Eingabe des CFREL Kommandos eine Relativ Record-Datei erstellen kann.

```
() CFREL

CREATE RELATIVE RECORD FILE
 PATHNAME:    VOL1.FILE2

 LOGICAL RECORD LENGTH:    80

 PHYSICAL RECORD LENGTH:    >230

 INITIAL ALLOCATION:    500

 SECONDARY ALLOCATION:    1000

 EXPANDABLE?: YES

 FORCED WRITE?: NO  YES

()
```

7.6 Anlegen sequentieller Dateien (CFSEQ Kommando)

Um eine sequentielle Datei anzulegen, gibt man das Kommando CFSEQ ein.

7.6.1 Format des CFSEQ Kommandos

CFSEQ
CREATE SEQUENTIAL FILE
PATHNAME: <acnm> @
LOGICAL RECORD LENGTH: [int]
PHYSICAL RECORD LENGTH: [int]
INITIAL ALLOCATION: [int]
SECONDARY ALLOCATION: [int]
EXPANDABLE?: <yes/no>
BLANK SUPPRESS?: <yes/no>
FORCED WRITE?: <yes/no>

7.6.2 Benutzerantworten beim CFSEQ Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Der Pfadname der sequentiellen Datei.
LOGICAL RECORD LENGTH:	O	Eine gerade ganze Zahl, die die durchschnittliche Satzlänge dieser Datei in Bytes angibt. Falls Leerzeichen im Satz unterdrückt werden sollen, sind diese bei der Satzlänge nicht mitzuzählen. Standardwert: 80

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PHYSICAL RECORD LENGTH:	O	Eine gerade Ganzzahl, die die Größe der Plattenübertragung in Bytes angibt. Der Standardwert wird zur Systemgenerierzeit angegeben.
INITIAL ALLOCATION:	O	Eine Ganzzahl, die die Anfangsgröße der Datei in logischen Sätzen angibt. Das System stellt einen Standardwert zur Verfügung, falls nichts angegeben wird.
SECONDARY ALLOCATION:	O	Eine Ganzzahl, die in logischen Sätzen den Zuwachs an Plattenspeicher angibt, welcher angelegt werden soll, falls der vorhandene Platz erschöpft ist. Wird kein Wert angegeben, gibt das System einen Wert vor.
EXPANDABLE?:	R	Y (yes) wird eingegeben, falls die Datei über die Anfangsgröße hinaus wachsen können soll. N (no) eingegeben, läßt diese Möglichkeit nicht zu.
BLANK SUPPRESS?:	R	Y (yes) wird eingegeben, falls jede Folge von Leerstellen in der Datei in verschlüsselter Form abgespeichert werden soll. N (no) wird eingegeben, falls jede Leerstelle einzeln abgespeichert werden soll. Bei einer Datei mit Leerstellenunterdrückung können keine Rewrite Operationen durchgeführt werden.
FORCED WRITE?:	R	Das Übertragen von der Platte auf sequentielle Medien erfolgt gewöhnlich erst dann, wenn Speicherplatz benötigt wird. Bei Eingabe von Y (yes) wird TAXO gezwungen sofort zu schreiben. Eingabe von N (no) erlaubt TAXO dagegen, dieses Übertragen von der Platte aufzuschieben.

7.6.3 Beispiel zum CFSEQ Kommando

Das folgende Beispiel zeigt, wie ein Benutzer mittels eines CFSEQ Kommandos eine sequentielle Datei erzeugen kann.

```
( ) CFSEQ
    CREATE SEQUENTIAL FILE
    PATHNAME: VOL1.FILE1
    LOGICAL RECORD LENGTH:
    PHYSICAL RECORD LENGTH:
    INITIAL ALLOCATION: 100
    SECONDARY ALLOCATION:
    EXPANDABLE?: YES
    BLANK SUPPRESS?: NO
    FORCED WRITE?: NO
( )
```

7.7 Erzeugen von Programm Dateien (CFPRO Kommando)

Durch das Kommando CFPRO (Create Program File) wird ein spezieller Typ einer Relativ Record Datei erzeugt, in welcher TAXO Program Images abgelegt werden. Gewöhnlich werden Programm Dateien expandierbar angelegt.

7.7.1 Format des Kommandos CFPRO

```
CFPRO
CREATE PROGRAM FILE
PATHNAME: <acnm> @
MAX NUMBER OF TASKS: <int>
MAX NUMBER OF PROCEDURES: <int>
MAX NUMBER OF OVERLAYS: <int>
INITIAL ALLOCATION: <int>
SECONDARY ALLOCATION: [int]
EXPANDABLE?: <yes/no>
```

7.7.2 Benutzerantworten auf das CFPRO Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME	R	Der Pfadname der Programm Datei
MAX NUMBER OF TASKS	R	Die erlaubte maximale Anzahl von Tasks in dieser Programm Datei. Standardwert = 255
MAX NUMBER OF PROCEDURES	R	Die erlaubte maximale Anzahl von Prozeduren in dieser Programmdatei. Standardwert = 255

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
MAX NUMBERS OF OVERLAYS	R	Die erlaubte max. Anzahl von Overlays in dieser Programmdatei. Standardwert = 255
INITIAL ALLOCATION	R	Eine gerade ganze Zahl, welche die Größe der ursprünglich erzeugten Datei in logischen Sätzen angibt. Der angezeigte Ursprungswert 85 gibt die minimale Größe für Verwaltungsinformationen an.
SECONDARY ALLOCATION	O	Eine gerade Ganzzahl, die die Größe der Datei in logischen Sätzen angibt, wenn die Datei vergrößert werden muß. Das System stellt einen Standardwert zur Verfügung, falls nichts angegeben wird.
EXPANDABLE?:	R	Y (yes) eingeben, erlaubt das Erweitern der Datei. N (no) erlaubt kein Erweitern. Standardwert ist YES.

7.7.3 Beispiel zum CFPRO Kommando

Nachfolgendes Beispiel zeigt die Eingabe eines CFPRO Kommandos, um eine Programm Datei zu erzeugen.

```
( ) CFPRO
    CREATE PROGRAM FILE
    PATHNAME: VOL1.DIR2.FILE2
    MAX NUMBER OF TASKS: 255
    MAX NUMBER OF PROCEDURES: 255
    MAX NUMBER OF OVERLAYS: 255
    INITIAL ALLOCATION: 1000
    SECONDARY ALLOCATION: 500
    EXPANDABLE?: YES
( )
```

7.8 Erzeugen von Image Dateien (CFIMG Kommando)

Zur Erzeugung einer Image Datei gibt man das Kommando CFIMG ein. TAXO unterstützt Image Dateien nur beim Roll-in/Roll-out, als System Lade Dateien und bei Dumps zur Diagnose. Dieser Dateityp eignet sich für keine andere Verwendung. Es wird nicht empfohlen, Image Dateien zu benutzen, weil sich bei Image Dateien die Satzgrenzen ändern, wenn die Datei von einer Platte auf eine Platte anderen Typs übertragen wird und die Daten durch übliche Techniken nicht wiederzugewinnen sind.

7.8.1 Format des CFIMG Kommandos

```
CFIMG
CREATE IMAGE FILE
PATHNAME: <acnm> @
ALLOCATION: <int>
```

7.8.2 Benutzerantworten beim CFIMG Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME:	R	Der Pfadname der Image Datei
ALLOCATION:	R	Eine gerade ganze Zahl, welche die Größe der Datei in logischen Sätzen angibt. Die logische Satzgröße ist durch die Sektorgröße der Platte bestimmt. Eine Abschätzung der benötigten Satzzahl könnte wie folgt erfolgen: (Objektlänge + Sektorgröße der Platte)/ Sektorgröße der Platte.

7.8.3 Beispiel zum CFIMG Kommando

Das folgende Beispiel zeigt die Eingabe eines CFIMG Kommandos, um eine Image Datei, die 1000 Sektoren umfassen soll, zu erzeugen.

```
( ) CFIMG
    CREATE IMAGE FILE
    PATHNAME: VOL1.FILE3
    ALLOCATION: 1000
( )
```

7.9 Löschen von Directories (DD Kommando)

Um ein Directory zu löschen, gibt man das Kommando DD (Delete Directory) ein. Löschen eines Directories bedeutet, das Directory mit allen darin enthaltenen Dateien (einschließlich der Subdirectories) aufzulösen. Mit anderen Worten, alle Dateien, die den Pfadnamen des aufgelösten Directories als Komponente führen, werden aufgelöst. Falls irgendeine Datei im angegebenen Directory oder eine Datei in einem Subdirectory vor Löschen geschützt ist, gibt das DD Kommando dies in einer Liste aus. Jede über einer vor Löschen geschützte Datei stehende Hierarchie wird nicht gelöscht. Das Kommando kann man abbrechen durch Drücken der ESC (escape) oder CMD (command) Taste. Es ist möglich, dieses Kommando sowohl im Vordergrund als auch im Hintergrund abzuarbeiten.

7.9.1 Format des DD Kommandos

```
DD
DELETE DIRECTORY
PATHNAME: <acnm> @
LISTING ACCESS NAME: [ acnm ] @
ARE YOU SURE?: <yes/no>
FOREGROUND (Y/N): <yes/no>
```

7.9.2 Benutzerantworten beim DD Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PATHNAME	R	Der Pfadname des zu löschenden Directories.
LISTING ACCESS NAME	O	Der Gerätename oder der Pfadname der Datei, auf welche(n) TAXO das Ergebnis der Löschung ausgeben soll. Standard ist der Terminal Local File.
ARE YOU SURE?:	R	Y (yes) zeigt an, daß sich der Benutzer sicher ist, dieses Directory zerstören zu wollen. N (no) hebt das Kommando wieder auf.
FOREGROUND (Y/N):	R	Y (YES) bedeutet, daß das Kommando im Vordergrund abgearbeitet werden soll.

7.9.3 Beispiel zum DD Kommando

Das folgende Beispiel zeigt den Aufbau eines Directories an. Um dieses Directory zu zerstören, wird in diesem Fall ein DD Kommando eingegeben.

```
( ) MD
    MAP DISK
    PATHNAME: .EXAMPLE
    LISTING ACCESS NAME: ME
    SHORT FORM?: YES
    TOP LEVEL ONLY?: NO
    DIRECTORY NODES ONLY?:NO
    FOREGROUND (Y/N): YES
    DISC MAP OF .EXAMPLE
```

```
TODAY IS 16:56:30 TUESDAY, JAN 17, 1978.

                                  NUMBER CURRENT TOTAL
                          FILE      OF    EOM   ALLOC
LV NAME                   TYPE   RECORDS  ADU    ADU        LAST UPDATE
-- ------------------      -      ------  -----  -----     ---------------
    EXAMPLE                D      12      4      4        1/17/78  16:54:35
 1 EXAMPLE:                                FILES=2  AVAILABLE=9
    DIR1                   D      24      3      3        1/17/78  16:56: 0
    FILE1                  S      0       0      1        1/17/78  16:54:35
 2  DIR1:                                 FILES=2  AVAILABLE=20
      FILE1                R      0       0      1        1/17/78  16:50:23
      FILE2                K             59     101       1/17/78  16:56: 2
**.VCATALOG.EXAMPLE.DIR1                  TOTAL SIZE = 110 ADUS
**.VCATALOG.EXAMPLE                       TOTAL SIZE = 115 ADUS
```

```
( ) DD
    DELETE DIRECTORY
    PATHNAME: .EXAMPLE
    LISTING ACCESS NAME: ME
    ARE YOU SURE?: YES
    FOREGROUND (Y/N): YES
    DELETED .EXAMPLE.DIR1.FILE1
    DELETED .EXAMPLE.DIR1.FILE2
    DELETED .EXAMPLE.DIR1
    DELETED .EXAMPLE.FILE1
    DELETED .EXAMPLE
( )
```

7.10 Löschen von Dateien (DF Kommando)

Um eine Datei zu löschen, benutzt man das DF (Delete File) Kommando. Ist die Datei ein Directory, so muß dieses leer sein, da sonst das Kommando nicht ausgeführt wird. Nichtleere Directories können nur durch das DD Kommando gelöscht werden. Ist die zu löschende Datei vor Löschen geschützt, so wird das Delete File Kommando nicht ausgeführt und ein Fehler angezeigt.

7.10.1 Format des DF Kommandos

```
DF
DELETE FILE
PATHNAME: <acnm> @
ARE YOU SURE?: <yes/no>
```

7.10.2 Benutzerantworten beim DF Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
PATHNAME	R	Man gibt den Zugriffsnamen der zu löschenden Datei ein.
ARE YOU SURE?:	R	Y (YES) zeigt an, daß sich der Benutzer sicher ist, dieses Directory zerstören zu wollen N (No) hebt das Kommando wieder auf.

7.10.3 Beispiel zum DF Kommando

```
( ) DF
    DELETE FILE
    PATHNAME: .JM
    ARE YOU SURE?:Y
( )
```

8 BATCH MODUS DES SYSTEM COMMAND INTERPRETERS

8.1 Allgemeines

Alle interaktiven Tasks laufen im Betriebssystem TAXO im Vordergrund Modus. Im Vordergrund Modus erwartet der SCI Eingaben vom Terminal und zeigt Ausgaben während der Verarbeitung am Terminal an. Eine Verarbeitung kann auch im Hintergrund erfolgen, doch erwartet hier der SCI keine Interaktionen mit dem Bildschirm. Welche SCI Kommandos wahlweise oder grundsätzlich einen Prozeß im Hintergrund (Background) starten, ist bei den jeweiligen Kommandos angegeben.

Der Benutzer kann jedes dieser Kommandos eingeben, um Hintergrundaktivitäten hervorzurufen. Gleichzeitig können andere Kommandos eingegeben und im Vordergrund ausgeführt werden. Hintergrundaktivitäten beeinflussen solche im Vordergrund nicht. Wird eine Hintergrundaktivität abgeschlossen, erscheint eine Ende-Meldung auf jedem Bildschirm, von dem die Aktivität angeregt wurde. Diese Ende-Meldung wird am Bildschirm erst nach Beendigung einer Vordergrundaktivität - bevor der SCI zur Eingabe des nächsten Vordergrund Kommandos durch ( ) auffordert - angezeigt. Die Meldung muß durch Drücken einer Taste (z.B. RETURN) quittiert werden.

In diesem Kapitel wird das Starten eines Batch Prozesses im Hintergrund mit dem XB (Execute Batch) Kommando und das Überwachen dieses Batch Prozesses beschrieben. Man gibt das XB Kommando ein, um den SCI im Hintergrund Modus zu aktivieren. Im Hintergrund Modus nimmt der SCI Kommandos von einer Eingabedatei oder einem Eingabegerät an, nicht jedoch vom Bildschirm. Alle TAXO Kommandos von diesem Eingabemedium (Batch Stream) müssen im Batch Format angegeben werden und somit alle Schlüsselwörter und Informationen über Parameter enthalten, die jenes Kommando erfordert.
Vom Terminal aus können keine zusätzlichen Informationen den Kommandos hinzugefügt werden. Zudem müssen die eingefügten Kommandos für die Ausführung im Hintergrund Modus geeignet sein. Die folgenden sind nicht geeignet für eine Batch Kommandofolge:

- Alle Debug und Text Editor Kommandos
- AT (Activate Task) Kommando
- KBT (Kill Background Task) Kommando
- MS (Modify Synonyms) Kommando

- SBS (Show Background Status) Kommando
- XD (Initiate Debug) Kommando
- XE (Execute Text Editor) Kommando
- XHT (Execute And Halt Task) Kommando

Zum Überwachen der Batch Abarbeitung stehen das SBS und KBT Kommando zur Verfügung, um den Status einer im Hintergrund ablaufenden Task anzeigen und abbrechen zu können. Das WAIT Kommando erlaubt es, auf die Beendigung eines Hintergrundprozesses zu warten. Darüberhinaus beschreibt dieses Kapitel das erforderliche Format für Kommandos, die in einen Batch Command Stream eingefügt werden sollen.

8.2 Ausführen des SCI im Batch Modus (XB Kommando)

Mit dem XB Kommando (Execute Batch) wird der System Command Interpreter im Hintergrund Modus aktiviert. Einmal aktiviert, läuft der SCI ohne jede Interaktion mit dem Terminal bis zum Ende. Danach gibt er eine Ende-Meldung auf jenem Terminal aus, von dem der Batch Lauf gestartet wurde. Ein Batch Command Stream wird mittels Text Editor erzeugt.

8.2.1 XB Kommando Format

XB
EXECUTE BATCH
INPUT ACCESS NAME: <acnm> @
LISTING ACCESS NAME: <acnm> @

8.2.2 Benutzerantworten zum XB Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
INPUT ACCESS NAME	R	Der Gerätename oder Pfadname der Datei, von welcher TAXO den Batch Command Stream lesen soll.
LISTING ACCESS NAME	R	Der Gerätename oder Pfadname der Datei, auf welche TAXO das Ergebnis der Ausführung des Batch Command Streams ausgeben soll. Dieses Gerät bzw. diese Datei dürfen nicht durch irgendein Kommando dieses Batch Command Streams angesprochen werden.

8.2.3 Beispiel zum XB Kommando

Das folgende Beispiel zeigt die Eingabe eines XB Kommandos zur Ausführung eines in einer Datei stehenden Batch Command Streams und zur Ausgabe des Resultats in eine Datei. Das Beispiel zeigt auch den Ausdruck einer Ende-Meldung (nicht Abbruch!) des Batchlaufes:

```
( ) XB
EXECUTE BATCH
    INPUT ACCESS NAME: MY.BATCH
    LISTING ACCESS NAME: MY.LIST
( ) <CR>
BATCH SCI    <XB> TERMINATED NORMALLY
( )
```

Das Symbol <CR> zeigt das Drücken der RETURN Taste an.

8.3 Zustandsanzeige von Tasks im Hintergrund (SBS Kommando)

Man gibt das SBS (Show Background Status) Kommando ein, um den Zustand eines gerade im Hintergrund ablaufenden Programmes zu erhalten, das vom Terminal des Benutzers aus gestartet wurde. Die nachfolgenden Zustandsmeldungen, welche Zustandsschlüssel von Tasks wiederspiegeln, werden durch das Kommando SBS erzeugt, falls der Hintergrund aktiv ist:

ACTIVE

AWAITING ANY I/O COMPLETION
(wartet auf Vervollständigung einer Ein-Ausgabe)

AWAITING ASSIGNMENT OF I/O DEVICE
(wartet auf die Zuweisung eines E/A-Gerätes)

AWAITING CHARACTER INPUT
(erwartet Eingabe von Zeichen)

AWAITING COMPLETION OF I/O
(erwartet Vervollständigung einer Ein-Ausgabe)

AWAITING COROUTINE ACTIVATION
(wartet auf Aktivierung einer Koroutine)

AWAITING DISK FILE UTILITY SERVICES
(wartet auf Unterstützung durch einen Plattendateidienst)

AWAITING FILE MANAGEMENT SERVICES
(erwartet Dienste des File Managements)

AWAITING GETMEM REQUEST
(erwartet Speicheranforderung)

AWAITING INITIAL LOAD
(erwartet INITIAL LOAD)

AWAITING TASK DRIVEN SVC PROCESSOR
(erwartet die Dienste eines SVC's)

AWAITING TERMINATION
(erwartet Beendigung)

ROLLING OUT
(Ausgelagert)

SUSPENDED
(Suspendiert)

SUSPENDED FOR INITIATE I/O THRESHOLD
(suspendiert, um eine Ein/Ausgabe zu initieren)

SUSPENDED FOR TIME DELAY
(für eine Zeitscheibe suspendiert)

Die folgenden Status Meldungen zeigen an, daß der Background nicht aktiv ist:

NO BACKGROUND ACTIVITY AT THIS TERMINAL
(keine Hintergrundaktivität an diesem Bildschirm)

TERMINATED
(beendet)

WAITING FOR EXECUTION TO BEGIN
(erwartet, mit der Ausführung zu beginnen)

SBS dient auch dazu, den Background Terminal Local File (TLF) anzuzeigen. Der TLF wird durch SBS angezeigt, wenn die Task im Hintergrund beendet ist, in den TLF durch die Background Task geschrieben wurde. (Befindet sich der SCI im Batch Modus, so wird der Background Terminal Local File in die Ausgabedatei oder auf das Ausgabegerät ausgegeben, wenn ein Background Kommando abgearbeitet ist).

Das Background TLF wird durch einen "open extend" Operationscode geöffnet, sodaß Einträge im TLF nicht verlorengehen, wenn der Benutzer das SBS zwischen den Bearbeitungen zweier Background Tasks einzugeben vergißt. Das WAIT Kommando (Kapitel 8.7) zeigt das Background TLF automatisch an, falls es existiert, wenn die Tasks im Background abgeschlossen wurden.

8.3.1 Beispiel zum SBS Kommando

Das folgende Beispiel zeigt die Eingabe eines XB Kommandos, um einen Batch Command Stream abzuarbeiten sowie die nachfolgende Eingabe von SBS Kommandos, um den Status des Background Programmes (in diesem Falle SCI) in verschiedenen Stadien der (Programm-) Ausführung zu erhalten:

```
( ) SBS
    SHOW BACKGROUND STATUS
    NO BACKGROUND ACTIVITY AT THIS TERMINAL
( )
```

```
( ) XB
    EXECUTE BATCH
    INPUT ACCESS NAME: MY.BATCH
    LISTING ACCESS NAME: MY.LIST
( )

( ) SBS
    SHOW BACKGROUND STATUS
    AWAITING FILE MANAGEMENT SERVICES
( )

( ) SBS
    SHOW BACKGROUND STATUS
    ACTIVE
( )

( ) SBS
    SHOW BACKGROUND STATUS
    AWAITING COROUTINE ACTIVATION
( )
```

8.3.2 STATE SYNONYM KEYWORD (Schlüsselwort)

STATE SYNONYM ist ein verstecktes Schlüsselwort, das nur innerhalb einer Verarbeitung oder im expert mode angegeben werden kann. Es erlaubt, den aktuellen Status des gerade aktiven Hintergrund Prozesses in einem Synonym abzulegen.
Ist ein Hintergrund Prozess nicht aktiv, wird dem Synonym der aktuelle Status (hier > FF) zugewiesen. Dies dient als Hilfe bei der Prüfung, ob ein Hintergrund Prozess aktiv ist, bevor ein anderer Hintergrund Prozess gestartet wird.
Diese Zuweisung erlaubt auch eine wirksame Fehlerbehandlung.
Beim Starten einer Hintergrund Verarbeitung während der aktiven Phase eines anderen Hintergrund Prozesses, beendet der SCI sofort die laufende sowie alle weiteren Verarbeitungen der betreffenden Station.
Das folgende SBS Kommando bewirkt, daß dem Synonym A der hexadezimale Wert wie >05 zugewiesen wird (Verzögerungszeit):

```
( ) SBS  STATE SYNONYM = A
```

8.4 Beenden von Tasks im Hintergrund

Das KBT Kommando wird eingegeben, um den Abbruch gerade im Hintergrund ablaufender Programme zu erzwingen.

8.4.1 Format des KBT Kommandos

KBT
KILL BACKGROUND TASK

8.4.2 Benutzerantworten auf das KBT Kommando

Keine Benutzerantworten erforderlich.

8.4.3 Beispiel zum KBT Kommando

Das folgende Beispiel zeigt die Eingabe des KBT Kommandos, um eine Task, die im Hintergrund abläuft, zu beenden, sowie die vom System ausgegebene Meldung.

```
( ) KBT
    KILL BACKGROUND TASK
    EXECUTION TERMINATED
( )
```

8.5 Format der Batch Kommandos

Gibt man im Betriebssystem TAXO Kommandos in Form eines Batch Command Streams ein, so muß jedes Kommando die folgenden Informationen enthalten, damit TAXO das Kommando abarbeiten kann:

- Das Kommando selbst
- Die Schlüsselwörter (prompting messages), welche mit dem Kommando verbunden sind
- Die Parameterwerte (Benutzerantworten), die zu den Schlüsselwörtern (command prompting messages) gehören.

Das Format, das man für diese Information braucht, lautet:

<Kommando> K1 = <Wert>, K2 = <Wert>, etc.

wobei

<Kommando> = die Zeichen des Kommandos selbst

K1, K2, etc. die Schlüsselwörter zu dem ersten, dem zweiten, und anderen Parametern dieses Kommandos.
Ein Schlüsselwort ist entweder die ganze Anfrage des Systems für jeden Parameter, oder eine Abkürzung der Systemfragen, die genug Zeichen enthält, um das Schlüsselwort zu bezeichnen. Ein Gleichheitsalgorithmus vergleicht die vollständigen Schlüsselwörter mit ihren Abkürzungen. Oftmals genügt es bereits, den ersten Buchstaben einzugeben.

Beispielsweise umfassen die Systemfragen eines Kommandos folgendes:

- SOURCE ACCESS NAME:
- OBJECT ACCESS NAME:
- LISTING ACCESS NAME:
- ERROR ACCESS NAME:
- OPTIONS:
- MACRO LIBRARY PATHNAME:

Ein Schlüsselwort für die erste Systemanfrage könnte S sein und für die zweite Meldung OB.

Der Gleichheitsalgorithmus arbeitet wie folgt:

Sei A ein Schlüsselwort.
Sei B eine Abkürzung eines Schlüsselwortes.

- Das Schlüsselwort A und das abgekürzte Schlüsselwort B müssen das gleiche erste Zeichen haben.
- Alle Zeichen in B müssen in A erscheinen und an gleicher Stelle wie in A stehen.

- Numerische Zeichen im Schlüsselwort A müssen auch im abgekürzten Schlüsselwort B erscheinen.
- Die erste Unstimmigkeit in B mit dem aktuellen Wort in A muß das erste Zeichen des nächsten Wortes in A sein.
- Zeichen nach dem ersten besonderen Zeichen in A werden ignoriert; z.B. (Blank) Keyword (Blank) entspricht Keyword.

<Wert> der Wert, der als Parameterwert anzugeben ist.

Zum Beispiel kann das Kommando in einem Batch Stream in einer Zeile eingegeben werden, wie folgt:

Kommandoname S = SOURCE, OB = OBJECT, L = LIST, OP = XREF

Die dazugehörige Prozedur würde im Hintergrund gestartet werden und die Batch Stream Bearbeitung auf ihre Vervollständigung warten. Die Vollständigkeitsmeldung erscheint im Batch Stream Listing nach der Auflistung der Schlüsselwörter und ihrer Werte.

8.6 Format des BATCH COMMAND STREAMS

Um unerwünschte Synonyme und Vorgabewerte zu entfernen, sollte das BATCH Kommando als erstes und das EBATCH Kommando als letztes Kommando in jedem Batch Stream stehen. Diese Kommandos entfernen unnötige vom SCI aus der Synonymliste des Benutzers generierte Synonyme. Das CM (Create Message) Kommando kann im Batch Modus genutzt werden, um Status Meldungen dem auslösenden Bildschirm zu übermitteln.

8.6.1 Format des BATCH Kommandos

Das BATCH Kommando hat folgendes Format:

BATCH (BEGIN BATCH EXECUTION),
USER-ID: [string]
STATION-ID: [string]
BATCH INPUT ACCESS NAME: [string]
BATCH LISTING ACCESS NAME: [string]
LS (LIST SYNONYMS)?: [yes/no (NO)]

Das BATCH Kommando wirkt nicht im Vordergrund Modus. Im Batch Modus, benötigt jeder dieser Parameter (außer LS) Werte, die von anderen Prozessoren benützt werden. LS ruft bei 'yes' das LS Kommando, welches alle Synonyme auf die Batch Listing File oder ein Ausgabegerät ausgibt.

Alle Parameter des BATCH Kommandos sind wahlfrei.

8.6.2 Format des EBATCH Kommandos

Das EBATCH Kommando hat folgendes Format:

EBATCH (END BATCH EXECUTION)
LS (LIST SYNONYMS? [YES/NO (NO)]
TEXT
CODE

Auch das EBATCH Kommando wirkt nicht im Vordergrund Modus. LS bewirkt bei 'YES' die Auflistung aller Synonyme auf der Batch Listing File. EBATCH übermittelt auch teilweise vollständige CFKEY Folgen im Batch Modus.

Mit Hilfe der Kommandos TEXT und CODE kann eine Meldung an den Bediener ausgegeben werden. Der Wert von CODE wird in Synonym $$BC in der Foreground TCA abgelegt.

8.7 Warten auf die Beendigung auf Backgroundaktivitäten (WAIT Kommando)

Endemeldungen von Background Kommandos werden gewöhnlich nicht angezeigt, außer nach Beendigung von Vordergrundkommandos oder falls durch SBS Kommando angefordert. Ist keine Vordergrundaktivität gewünscht, kann man das WAIT-Kommando eingeben. Damit erscheinen die Background-Meldungen, ohne daß dauernd SBS eingegeben werden muß. Das WAIT-Kommando hat keine Parameter. WAIT gibt folgende Meldung aus bis die Task im Hintergrund beendet ist:

-WAITING FOR BACKGROUND TASK TO COMPLETE-

Um das WAIT-Kommando zu tilgen und im Vordergrund weiterarbeiten zu können, drücke man die COMMAND Taste. Das WAIT-Kommando zeigt die Background Terminal Local File an (wenn vorhanden), sobald die Background Task beendet ist.
Ein neues verstecktes Keywort MESSAGE erlaubt es, die Endemeldung eines Hintergrundprozesses zu unterdrücken. Damit werden Kommandofolgen wie z.B.

XB . . .
WAIT MESSAGE = NO
Q

möglich, was ein automatisches Logoff nach Beendigung von zwei Batch Streams bewirkt (d.h. der Benutzer braucht nicht auf das Ende der Batch Ausführung zu warten). Sinnvoll ist dies allerdings nur innerhalb von Foreground-Kommandoprozeduren.

8.8 Zählen von Fehlern in Batchstreams (EC-Kommando)

Tritt bei der Abarbeitung eines Kommandos ein Fehler auf, so wird von dem jeweiligen Dienstprogramm das Completion Code Synonym $$CC i.a. auf einen Wert ungleich Null gesetzt. Mit Hilfe des EC Kommandos kann dieser Wert geprüft werden. Ist er ungleich Null, wird von EC der Wert des Fehlerzählsynonyms EC um 1 erhöht. Der Wert von EC kann am Ende des Batch Streams durch einen geeigneten Aufruf von CM (Create Message) am Benutzerterminal ausgegeben werden. Das Synonym EC muß natürlich am Anfang des Batch Streams auf 0 gesetzt werden.

Beispiel eines Batch Streams mit EC Kommandos:

```
BATCH LS=Y
AS SYN=$E$C, V=0
    .
    .
Kommando
EC
Kommando
EC
    .
    .
CM R=ME, T="BATCH BEENDET.-@$E$C FEHLER AUFGETRETEN"
EBATCH
```

Durch diesen Batch Stream wird am Ende die Zahl der aufgetretenen Fehler am Benutzerterminal angezeigt.

8.9 Expert Mode

Die Kommandoeingabe im Expert Mode ist vergleichbar mit dem Kommandoaufruf in Batch-Kommandofolgen, d. h., ein Kommando kann interaktiv eingegeben werden, indem der Kommandoname, gefolgt von einem Zwischenraum und dann alle Keywords mit den ihnen zugeordneten Werten, eingetastet wird. Dabei kann das letzte Zeichen einer Zeile ein Fortsetzungszeichen sein ("," oder "="), was bedeutet, daß der Kommandoaufruf durch weitere Eingaben fortgesetzt wird.

ANMERKUNG:

Wortzuweisungen von Keywords sind nur für das gerufene Kommando möglich. Werden während der Kommandoabarbeitung Unterprozeduren aufgerufen, so werden Wortzuweisungen an Keywords der Unterprozeduren nicht automatisch übergeben.

Ist ein Kommandoaufruf im Expert Mode vollständig eingegeben, gibt der SCI am Bildschirm die Keyword Maske aus, wobei jedoch die erste Zeile, die normalerweise den vollen Kommandonamen enthält, unterdrückt wird. Würden bei der Kommandoeingabe alle Pflichteingaben spezifiziert und sind sie syntaktisch richtig, wird das Kommando sofort ausgeführt. Ein Beispiel soll die Kommandoeingabe im Expert Mode verdeutlichen:
Wurde für ein Keyword, bei dem eine Pflichteingabe besteht, kein Wert angegeben und existiert auch keine Vorbesetzung dafür, wird für dieses Keyword in der Keyword Maske eine Eingabe angefordert und erst dann die Kommandoausführung gestartet.

```
( ) SF FILE = .S$NEWS
```

Durch diese Eingabe wird die Datei .S$NEWS am Benutzerterminal angezeigt, ohne daß der SCI eine Eingabe für das Keyword File PATHNAME des SF Kommandos anfordert. Es sei darauf hingewiesen, daß Keywords wie in Batches eindeutig abgekürzt werden können (im Beispiel: FILE PATHNAME wird abgekürzt durch FILE). Würde nun das SF Kommando erneut aufgerufen, hätte das Keyword FILE PATHNAME die Vorbesetzung .S$NEWS. Um diese Datei jedoch erneut anzuzeigen, kann der Benutzer jetzt auch das SF Kommando, gefolgt von einem Punkt, eintasten:

```
( ) SF.
```

Die Datei .S$NEWS wird wieder angezeigt. Daraus folgt, wenn bei der Kommandoeingabe hinter dem Kommando ein Punkt eingegeben wird, nimmt der SCI an, daß der Benutzer ein Expert Mode arbeitet. Die Zuweisung FILE = .S$NEWS im ersten Beispiel und der Punkt "." im zweiten Beispiel veranlassen den SCI im Expert Mode zu arbeiten.

Der Expert Mode im SCI erlaubt also eine schnellere Kommandoeingabe, wenn alle Keywords bereits vorbesetzt sind oder mit Standardwerten für optimale Parameter gearbeitet werden soll. Eine weitere wichtige Funktion besteht darin, sog. "versteckte" Keywords mit Werten zu versorgen. Versteckte Keywords sind solche, die zwar für ein Kommando definiert sind, in der Keyword Maske jedoch nicht erscheinen, weil für sie kein Typ festgelegt ist.

Eine weitere Eigenschaft des Expert Modes besteht darin, daß Keywords, die beim Kommandoaufruf zwar angegeben wurden, denen aber kein Wert zugewiesen wurde, automatisch den Wert "Y" (für Yes) erhalten.

Beispiel:

Das SCI Kommando DCOPY besitzt ein verstecktes Keyword WITHOUT DEFAULTS. Hat dieses Keyword den Wert "Y", wird der Save Mode von DCOPY aktiviert, andernfalls der Standardkopiervorgang. Ein Aufruf des Save Modes von DCOPY kann nun auf eine der folgenden Arten erfolgen:

```
( ) DCOPY WITHOUT DEFAULTS = YES
( ) DCOPY W = YES
( ) DCOPY WITHOUT DEFAULTS
( ) DCOPY W
```

Diese Eigenschaft des Expert Modes sollte allerdings nur mit Vorsicht angewandt werden,und nur dann, wenn der Benutzer mit der Funktion des Kommandos vertraut ist.

Eine weitere Eigenschaft des Expert Modes ist es, "versteckten" Prompts beim Kommandoaufruf Werte zuzuweisen. Versteckte Prompts sind solche, für die in der Prozedurdefinition kein Typ festgelegt ist. Solche versteckten Prompts können spezielle Funktionen eines Kommandos veranlassen, die für den Benutzer normal nicht notwendig sind. Als Beispiel hierfür sei die DCOPY Prozedur genannt:

```
DCOPY (DISK COPY AND RESTORE),
        WITHOUT DEFAULTS
.IF &WITHOUT DEFAULTS ,GE, "Y"
        Bid DCOPY Task for Save Mode
.ELSE
        Bid DCOPY Task for Copy Mode
.ENDIF
```

Wird dieses Kommando (wie allgemein üblich) mit

```
( ) DCOPY
```

aufgerufen, ist für das Prompting WITHOUT DEFAULT kein Wert vorhanden (d. h. &WITHOUT DEFAULTS ist kleiner als "Y") und die Kommandos hinter .ELSE werden ausgeführt. Erfolgt der Aufruf jedoch mit

```
( ) DCOPY WITHOUT DEFAULTS = YES
```

so ist die .IF Bedingung erfüllt und die Kommando vor .ELSE werden ausgeführt.

9 ZUWEISUNG UND KONTROLLE EINER LOGISCHEN EINHEIT

9.1 Allgemeines

Anwendungsprogramme unter TAXO brauchen keine spezifischen Geräte oder Dateien anzugeben, auf die ausgegeben wird oder von denen Eingaben kommen. Stattdessen spezifizieren sie logische Einheitennummern (LUNO's), um Ein- und Ausgabegeräte zu repräsentieren. Eine LUNO wird einem Gerät oder einer Datei durch ein LUNO Zuweisungskommando eines Benutzers oder eines ausführenden Programmes zugewiesen. Die Arten von LUNO Zuweisungen, die ausgeführt werden können, sind folgende:

Typ der LUNO Zuweisung	Zuweisungsbereich
Global	Wenn eine Datei oder ein Gerät für Ein-/Ausgabeoperationen an mehreren Terminals verfügbar sein muß, wird eine globale LUNO durch ein AGL Kommando (globale LUNO Zuweisung) zugewiesen. Ein Gerät, auf das eine globale LUNO zugewiesen ist, ist von jedem Terminal für Ein-Ausgabe Operationen zugänglich.
Terminal-Local	Wenn eine Datei oder ein Gerät für Ein-/Ausgabe Operationen nur an einem Terminal verfügbar sein muß, wird eine Terminal-Local LUNO durch ein AL Kommando (LUNO Zuweisung) zugewiesen.
Task-Local	Ausführungsprogramme können LUNO Zuweisungen für ihre eigenen Ein-/Ausgabe Operationen benötigen. Diese Task-Local LUNO Zuweisungen erfordern keine Kommandoeingabe durch einen Benutzer.

LUNO Zuweisungen gibt es entsprechend den Notwendigkeiten und der Effizienz eines jeden Programms.

Die folgenden Kommandos sind vorhanden, um globale und terminal-locale LUNO Zuweisungen durchführen zu können:

- AGL (Assign Global LUNO - globale LUNO Zuweisung)
- AL (Assign LUNO - terminal locale LUNO Zuweisung)

Die folgenden Kommandos sind vorhanden, um globale und terminal-locale Zuweisungen aufzuheben:

- RGL (Release Global LUNO - Aufhebung einer globalen LUNO Zuweisung)
- RL (Release LUNO - Aufhebung einer terminal-localen LUNO Zuweisung)
- RAL (Release all Local LUNO's - Aufheben aller terminal-localen Zuweisungen)

Task-Local LUNO's werden durch die Task aufgehoben oder, wenn dies nicht geschieht, werden sie automatisch aufgehoben, wenn die Task beendet wird.

9.2 Zuweisung einer globalen logischen Einheit (AGL Kommando)

Benutzer geben ein AGL Kommando (globale LUNO Zuweisung - Assign Global LUNO) ein, um eine LUNO auf ein Gerät oder eine Datei zuzuweisen, die an mehr als einem Terminal verfügbar sein muß. Wird eine schon belegte globale LUNO zugewiesen, so wird ein Fehler gemeldet. Das System wählt eine unbenutzte globale LUNO aus, wenn keine LUNO Eingabe als Antwort auf die LUNO Systemmeldung erfolgt ist. In jedem Falle wird die folgende Meldung am Terminal angezeigt:

ASSIGNED LUNO = XX

wobei XX die LUNO bezeichnet, die zugewiesen wurde.

9.2.1 Format des AGL Kommandos

AGL
ASSIGN GLOBAL LUNO
LUNO: [int]
ACCESS NAME: < acnm > @
PROGRAM FILE?: [yes/no]
DISPLAY ? [yes/no]

9.2.2 Antwort des Benutzers zum AGL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO	O	Eine logische Einheitennummer im Bereich von 0 bis FF_{16} kann angegeben werden. Siehe die Liste der vom System reservierten globalen LUNO Zuweisungen in Anhang C. Vom System reservierte LUNO Zuweisungen dürfen NICHT verwendet werden, da sonst das System zusammenbricht oder nicht funktionsfähig ist. Wenn nichts eingegeben wird, generiert das System eine unbenutzte, reservierte, globale LUNO Zuweisung = Standardwert.
ACCESS NAME	R	Der Zugriffsname eines physikalischen Gerätes oder der Pfadname einer Datei, auf den die LUNO zugewiesen werden soll.
PROGRAM FILE?:	R	Die Eingabe N (no) bedeutet, daß die LUNO einer sequentiellen, relativen oder einer Key Index Datei zugewiesen wird. Die Eingabe Y (yes) bedeutet, daß die LUNO einer Programm, einer Directory oder einer Image Datei zugewiesen wird. Standardwert ist N (no).
DISPLAY	O	Mit Hilfe des versteckten Keywords vom Typ YES/NO kann die Meldung "Assigned Luno = > nn" unterdrückt werden.

9.2.3 Beispiel zum AGL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein AGL Kommando eingeben kann, um einen Zeilendrucker (LP04) einer globalen LUNO (30_{16}) zuzuweisen:

```
( ) AGL
    ASSIGN GLOBAL LUNO
    LUNO: >30
    ACCESS NAME: LP04
    PROGRAM FILE?: NO
    ASSIGNED LUNO = >30
( )
```

9.3 Aufheben einer globalen LUNO (RGL Kommando)

Um ein Gerät oder eine Datei von einer globalen LUNO Zuweisung zu trennen, gibt der Benutzer das RGL Kommando (Release Global LUNO) ein. Globale LUNO's, die gerade I/O-Operationen ausführen, können nicht aufgehoben werden. Globale LUNO Zuweisungen, die für das System reserviert sind, dürfen NICHT aufgehoben werden (siehe dazu Anhang C).

9.3.1 Format des RGL Kommandos

```
RGL
RELEASE GLOBAL LUNO
LUNO: <int>
```

9.3.2 Antworten des Benutzers zum RGL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO	R	Die Nummer einer logischen Einheit, von der die globale Zuweisung eines Gerätes oder einer Datei aufzuheben ist.

9.3.3 Beispiel für ein RGL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein RGL Kommando eingeben muß, um ein Gerät oder eine Datei von einer globalen Zuweisung einer LUNO (25_{16}) zu trennen:

```
( ) RGL
    RELEASE GLOBAL LUNO
    LUNO: >25
( )
```

9.4 Zuweisung einer Terminal-Local logischen Einheit (AL Kommando)

Benutzer geben ein AL Kommando (Assign LUNO) ein, um einem Gerät oder einer Datei eine LUNO zuzuweisen, die für Ein-/Ausgabeoperationen auf ihrem Terminal zugänglich sein muß. Wird ein Gerät oder eine Datei einer schon belegten terminal-local LUNO zugewiesen, so wird ein Fehler gemeldet. Das System wählt eine unbenutzte terminal-local LUNO aus, wenn keine LUNO Eingabe als Antwort zu der Systemmeldung "LUNO" erfolgt. In jedem Falle wird die folgende Meldung auf dem Bildschirm angezeigt:

```
ASSIGNED LUNO = XX
```

wobei XX die LUNO bezeichnet, die zugewiesen wurde.

ANMERKUNG:
Benutzer müssen beachten, daß sie keine für SCI und Sprachen reservierte LUNO's zuweisen: 01, 02, 03> 10, und >E0.

9.4.1 Format des AL Kommandos

AL
ASSIGN LUNO
LUNO: [int]
ACCESS NAME: <acnm > @
PROGRAM FILE?: [yes/no]
DISPLAY? [yes/no]

9.4.2 Antworten des Benutzers zum AL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	O	Eine logische Nummer einer Einheit im Bereich von 0 bis FF_{16}. Wenn nichts angegeben wird, generiert das System eine unbenutzte terminal locale LUNO.
ACCESS NAME:	R	Ein gültiger Gerätename eines physikalischen Gerätes oder der Pfadname einer Datei, auf den die LUNO zugewiesen werden soll.
PROGRAM FILE?:	R	Eingabe von Y(ES) ist erforderlich, wenn die LUNO auf eine Directory, Image oder Programm Datei zugewiesen werden soll. Bei anderen Dateitypen ist N(O) einzugeben. Standardwert = NO.
DISPLAY ?	O	Mit Hilfe des versteckten Keywords vom Typ YES/NO kann die Meldung "Assigned LUNO = >nn" unterdrückt werden.

9.4.3 Beispiel zum AL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein AL Kommando eingeben kann, um einen Zeilendrucker (LP04) einem terminal-local LUNO (30_{16}) zuzuweisen.

```
( )     AL
        LUNO: >30
        ACCESS NAME: LP04
        PROGRAM FILE?: NO
        ASSIGNED LUNO = >30

( )
```

9.5 Aufheben einer Terminal-Local LUNO (RL Kommando)

Um ein Gerät, das zum Terminal des Benutzers gehört, von einer LUNO Zuweisung zu entbinden, gibt der Benutzer das RL Kommando (Release LUNO) ein.

ANMERKUNG:
Eine momentan aktive LUNO kann nicht aufgehoben werden.

9.5.1 Format des RL Kommandos

RL
RELEASE LUNO
LUNO: <int>

9.5.2 Antworten des Benutzers beim RL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	R	Die logische Nummer einer Einheit, von welcher die terminal-local Zuweisung eines Gerätes oder einer Datei aufzuheben ist.

9.5.3 Beispiel zum RL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein RL Kommando eingeben kann, um ein Gerät von einer terminal-local LUNO (25_{16}) Zuweisung aufzuheben.

```
( ) RL
    RELEASE LUNO
    LUNO: >25
( )
```

9.6 Aufheben aller Terminal-Local LUNO Zuweisungen (RAL Kommando)

Benutzer geben das RAL Kommando (Release ALL LUNO's) ein, um alle LUNO Zuweisungen, die zum Terminal des Benutzers gehören, aufzuheben.

9.6.1 Format des RAL Kommandos

RAL
RELEASE ALL LOGICAL UNITS
ARE YOU SURE?: <yes/no>

9.6.2 Antworten des Benutzers beim RAL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
ARE YOU SURE?:	R	Hier muß der Benutzer angeben, ob er sich sicher ist, alle terminal-local LUNO Zuweisungen aufheben zu wollen (Eingabe = YES). Bei Eingabe von N(O) wird das Kommando abgebrochen.

9.6.3 Beispiel zum RAL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein RAL Kommando eingeben muß, um alle Geräte, die zum Terminal des Benutzers gehören, von LUNO Zuweisungen zu entbinden.

```
( ) RAL
    RELEASE ALL LOCAL LUNOS
    ARE YOU SURE?: YES
( )
```

9.7 Vorwärtspositionieren logischer Einheiten (FL Kommando)

Das Betriebssystem greift auf Informationen von sequentiellen Dateien und Geräten aufgrund einer Satz-für-Satz-Basis zu. Benutzer können die augenblickliche Datenzugriffsposition um einen oder mehrere Sätze durch Eingabe eines FL Kommandos (Forwardspace LUNO) vorwärts bewegen.

9.7.1 Format des FL Kommandos

FL
FORWARD SPACE LUNO
LUNO: <int>
RECORD COUNT: <int>

9.7.2 Antworten des Benutzers beim FL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	R	Die logische Einheitennummer eines sequentiellen Gerätes oder Files, das vorwärts positioniert werden soll.
RECORD COUNT:	R	Die Anzahl von Sätzen (von 1 bis 65535), um welche die sequentielle Datei oder das Gerät vorwärts positioniert werden soll. Stößt TAXO während des Vorwärtslaufs auf das Ende der Datei (EOF), so positioniert TAXO die sequentielle Datei oder das Gerät auf den dem EOF folgenden Satz.

9.7.3 Beispiel zum FL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein FL Kommando eingeben muß, um eine mit LUNO 32 zugewiesene Magnetbandeinheit um 3 Sätze vorwärts zu bewegen:

```
( ) FL
    FORWARD SPACE LUNO
    LUNO: 32
    RECORD COUNT: 3
( )
```

9.8 Rücksetzen einer logischen Einheit (BL Kommando)

Das Betriebssystem greift auf Informationen von sequentiellen Dateien und Geräten aufgrund einer Satz-für-Satz-Basis zu. Benutzer können die augenblickliche Datenzugriffsposition durch die Eingabe eines BL Kommandos (Backspace LUNO) um einen oder mehrere Sätze rückwärts positionieren.

9.8.1 Format des BL Kommandos

BL
BACKSPACE LUNO
LUNO: <int>
RECORD COUNT: <int>

9.8.2 Antworten des Benutzers beim BL Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	R	Die logische Einheitennummer eines sequentiellen Gerätes oder einer Datei, die rückwärts positioniert werden soll.
RECORD COUNT:	R	Die Anzahl von Sätzen (von 1 bis 65535), um welche eine sequentielle Datei oder ein Gerät in Rückwärtsrichtung eingestellt werden soll. Wird von TAXO das Ende der Datei (EOF) vor Beginn des Rücksetzens erkannt, so wird das sequentielle Gerät oder die Datei auf den Anfang des ersten Satzes der Datei oder des Gerätes positioniert.

9.8.3 Beispiel zum BL Kommando

Das folgende Beispiel zeigt, wie ein Benutzer ein BL Kommando eingeben muß, um auf einer Magnetbandeinheit, die mit LUNO 32 zugewiesen ist, einen Rücklauf um 4 Sätze zu erhalten:

```
( ) BL
    LUNO: 32
    RECORD COUNT: 4
( )
```

9.9 Zurückspulen logischer Einheiten (RWL Kommando)

Benutzer geben das RWL Kommando (Rewind Logical Unit) ein, um eine sequentielle Datei oder ein Gerät zurückzuspulen. Wird eine Datei oder ein Gerät zurückgespult, so wird die augenblickliche Datenzugriffsposition von TAXO auf den ersten Satz dieser Datei oder dieses Gerätes positioniert. TAXO ignoriert die Eingabe eines RWL Kommandos, wenn das Gerät oder die Datei nicht zurückgespult werden kann.

9.9.1 Format des RWL Kommandos

```
RWL
REWIND LOGICAL UNIT
LUNO: <int>
```

9.9.2 Antwort des Benutzers beim RWL Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	R	Die logische Nummer einer Datei oder eines Gerätes, welches zurückgespult werden soll.

9.9.3 Beispiel zum RWL Kommando

Das folgende Beispiel zeigt, wie ein Bediener ein RWL Kommando eingeben muß, um die mit LUNO 27 bezeichnete Datei zurückzuspulen:

```
( ) RWL
    LUNO: 27
( )
```

9.10 Anzeigen des I/O Status einer logischen Einheit (SIS Kommando)

Bediener können sich den Ein-/Ausgabestatus globaler, terminal-localer und task-localer (logischer) Einheiten durch Eingabe des SIS Kommandos (Show I/O Status) anzeigen lassen. Der Status einer LUNO oder aller LUNO's eines Typs (global, terminal-local oder task-local) kann angezeigt werden. Tabelle 9-1 beschreibt die Überschriften, unter welchen die Statusinformationen angezeigt werden können (nicht alle Überschriften werden von allen LUNO Typen gebraucht).

9.10.1 Format des SIS Kommandos

```
SIS
SHOW I/O STATUS
LUNO: <int>
TYPE (TASK, STATION, GLOBAL,A): <string>
TASK RUN ID: <int> [ wird nur angezeigt, wenn Typ TASK angegeben wurde ]
```

Tabelle 9-1 Informationen, die aufgrund eines SIS Kommandos angezeigt werden

Überschrift	Angezeigte Information
TYPE	Der Status einer Luno: S (Terminal-Lokal) T (Task-Lokal) G (Global)
LUNO	Die LUNO, deren Status gewünscht wurde oder alle LUNO's eines Typs, deren Status gewünscht wurde.
DEV. NAME	Der der LUNO zugewiesene Gerätename (wurde eine Datei der LUNO zugewiesen, so bezeichnet der Gerätename das Gerät, auf dem die Datei liegt).
TSK ID	Benutzt eine Task momentan eine LUNO, so wird die Laufzeit Kennummer der Task gelistet, die die LUNO eröffnet hat.
STN.NO	Die Nummer des Terminals, mit welchem eine terminal-local LUNO verbunden ist.
OPN? Y/N	Y (yes) bedeutet, wenn die LUNO gerade durch eine Task eröffnet ist. N (no) bedeutet, daß die LUNO momentan nicht eröffnet ist.
FILE TYPE	Eine der folgenden Dateitypen wird gelistet für die Dateien, auf die LUNOs zugewiesen sind: . SEQ (Sequential File - sequentielle Datei) . REL (Relative Record File - relative Datei) . PGM (Program File - Programm Datei) . IMG (Image File - Image File) . DEV (Device - LUNO ist auf ein Gerät zugewiesen) . KEY (Key Indexed File - Key Index Datei) . DIR (Directory File - Directory Datei)
FILE NAME	Der Zugriffsname, auf den die LUNO zugewiesen ist.

9.10.2 Antworten beim SIS Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	O	Die logische Nummer einer Einheit, deren Status gewünscht wird. Wird LUNO nicht eingegeben, so werden alle LUNO's des angegebenen Typs angezeigt.
TYPE: (TASK, STATION, GLOBAL,A)	R	Eine Zeichenkette (GLOBAL, STATION, TASK oder ALL), die den Typ der LUNO bezeichnet, deren Status gewünscht wird. Benutzer können die Zeichenkette durch Eingabe eines Zeichens abkürzen, das eindeutig die Zeichenkette identifiziert. So eignet sich für die Eingabe z.B. besonders ein G oder ein S für einen globalen oder terminal-local LUNO. Die Antwort 'A' gibt alle zugewiesenen LUNO's aus.

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
TASK RUN ID:	R	Benutzer geben eine Task-Nummer nur dann ein, wenn sie den Status einer task-local LUNO anzeigen lassen wollen. Die eingegebene Nummer muß die vom System zugewiesene Laufzeit Kennummer sein, wenn die Task gerade ausgeführt wird.

9.10.3 Beispiel zum SIS Kommando

Das folgende Beispiel zeigt, wie ein Benutzer den Status aller globalen LUNOs anzeigen kann:

```
()SIS
SHOW I/O STATUS
 LUNO:
TYPE (TASK,STATION,GLOBAL,A): S  A
TYPE   DEV.  TSK  ST  OPN  FILE  FILE
* LUN  NAME   ID  NO  Y/N  TYPE  NAME
T 04   DS01   6B   1   Y   SEQ.  PAE5.S$PROC.SIS
S 00   ST01           N   DEV.
G 09   DS01           N   DEV.
G 07   DS01           N   PGM.  PAE5.DORIS.PROGB
G 14   ST01           N   DEV.
G 15   ST01           N   DEV.
G 50   ST04           N   DEV.
G 03   DS01           N   REL.  PAE5.S$TCALIB
G 02   DS01           N   REL.  PAE5.S$BGTCA
G 01   DS01           N   REL.  PAE5.S$FGTCA
G 10   DS01           N   PGM.  PAE5.S$SDS$
G 02   DS02           N   DEV.
G 03   DS03           N   DEV.
G 00   ST01           N   DEV.
G 0A   DS01           N   REL.  PAE5.S$OVLYA
G 0D   DS01           N   PGM.  PAE5.S$PROGA
G 0B   DS01           N   PGM.  PAE5.S$PROGA
G 0F   DS01           N   PGM.  PAE5.S$PROGA
G 0E   DS01   00      N   IMG.  PAE5.S$ROLLA
()  0
```

ANMERKUNG:
Stationsnummern erscheinen nur, wenn der Status von task-local LUNO's gefordert wird.

9.11 Modifizieren des Schutzes einer globalen LUNO gegen Freigeben (MLP Kommando)

Mit dem MLP (Modify LUNO Protection) kann eine globale LUNO gegen Freigabe durch ein RGL (Release Global LUNO) Kommando geschützt oder dieser Freigabeschutz wieder aufgehoben werden. Ein Freigabeschutz ist für solche globalen LUNOs sinnvoll, die ab ihrer Zuweisung mit dem AGL (Assign Global LUNO) Kommando bis zum Ausschalten des Systems zur Verfügung stehen müssen und während dieser Zeit nicht verändert werden dürfen.

Beim Initialisieren des Systems mit dem IS (Initialize System) Kommando werden alle vom TAXO Betriebssystem verwendeten globalen LUNOs gegen Freigeben geschützt. Zu diesem Zeitpunkt können auch die für Anwendungsprogramme notwendigen globalen LUNOs zugewiesen und gegen Freigeben geschützt werden. Es ist jedoch auch zu einem beliebigen späteren Zeitpunkt möglich, eine zugewiesene LUNO gegen Freigeben zu schützen oder diesen Schutz aufzuheben.

> VORSICHT:
> Ein Aufheben des Freigabeschutzes der im Anhang B aufgeführten System LUNOs mit anschließender Freigabe kann zu nicht vorhersehbaren Fehlern und Systemabstürzen führen.

9.11.1 Format des MLP Kommandos

MLP
MODIFY LUNO PROTECTION
PROTECT OR UNPROTECT? (P/U):<string>
LUNO: <integer>

9.11.2 Antworten des Benutzers beim MLP Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
PROTECT OR UNPROTECT? (P/U)	R	Soll die LUNO gegen Freigeben geschützt werden, so ist P (= Protect) einzugeben. Soll der Freigabeschutz aufgehoben werden, so ist U (= Unprotect) einzugeben.
LUNO	O	Angabe der LUNO, für die der Freigabeschutz modifiziert werden soll. Wird keine Eingabe gemacht, so wird die am Benutzerterminal zuletzt zugewiesene globale LUNO verwendet.

9.11.3 Beispiel zum MLP Kommando

Das folgende Beispiel zeigt die Eingabe des MLP Kommandos, um die globale LUNO 10 zu schützen.

```
( ) MLP
MODIFY LUNO PROTECTION
PROTECT OR UNPROTECT? (P/U): P
LUNO: 10
( )
```

Das folgende Beispiel zeigt die Eingabe des MLP Kommandos, um den Freigabeschutz der letzten am Benutzerterminal zugewiesenen globalen LUNO aufzuheben. Die zuletzt zugewiesene globale LUNO, die von TAXO ausgewählt wird, ist in diesem Beispiel LUNO >7.

```
( ) AGL
ASSIGN GLOBAL LUNO
    LUNO:
    ACCESS NAME: LP01
ASSIGNED LUNO = >7

( ) MLP
MODIFY LUNO PROTECTION
    PROTECT OR UNPROTECT? (P/U):U
    LUNO: >7
```

9.12 WRITE END-OF-FILE TO LUNO (WEOF Kommando)

Im Betriebssystem TAXO kann der Benutzer ein EOF Zeichen (Bandmarke) an die momentane Zugriffsposition eines Magnetbandes setzen.

9.12.1 Format des WEOF Kommandos

```
WEOF
WRITE EOF TO LUNO
LUNO : <int>
```

9.12.2 Benutzerantworten zum WEOF Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
LUNO:	R	Die logische Nummer der sequentiellen Datei oder des Gerätes, worauf die EOF Marke geschrieben werden soll.

9.12.3 Beispiel zum WEOF Kommando.

Das folgende Beispiel zeigt, wie der Benutzer das WEOF Kommando eingeben muß, um eine EOF Marke auf ein Magnetband zu schreiben. (Zuweisung auf LUNO 32)

```
[ ] WEOF
WRITE EOF TO LUNO
LUNO : 32
```

10 SONSTIGE KOMMANDOS

10.1 Allgemeines

In diesem Kapitel werden Kommandos beschrieben, die verschiedene Systemstatusanzeigen ermöglichen und Meldungen ins Systemlogbuch und an andere Terminals senden. Außerdem sind am Ende noch einige allgemeine Kommandos beschrieben.

10.2 Anzeige von Systemdatum und Systemzeit - SDT Kommando

Um auf einem Terminal oder in einem Batch Listing das aktuelle Datum und die Zeit anzuzeigen, gibt man das SDT-Kommando (Show Date and Time) ein. Die Uhrzeit wird im 24-Stunden Modus angegeben mit dem Stundenwert zwischen 00 und 23. Zum Beispiel wird 2.13 h Nachmittags als 14.13 angezeigt und 2.13 h morgens als 02.13 h.

10.2.1 Format des SDT-Kommandos

SDT
SHOW DATE AND TIME

10.2.2 Benutzerantworten beim SDT-Kommando

Es werden keine Antworten erwartet.

10.2.3 Beispiel zum SDT-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das SDT-Kommando eingeben kann, um das aktuelle Systemdatum und die Zeit zu erhalten:

```
( ) SDT
SHOW DATE AND TIME
13:34:10 SATURDAY, FEB 05, 1977.:
( )
```

10.3 Senden einer Nachricht zu einem Terminal - CM Kommando

Man kann eine einzeilige Nachricht auf ein anderes Terminal senden, indem man das CM (Create Message) Kommando eingibt. Das empfangende Terminal kann durch ein CM-Kommando nur dann eine Nachricht erhalten, wenn das Folgende zutrifft:

- Ein Benutzer arbeitet nach Log-in am Terminal (sofern Log-in erforderlich ist).
- Das Terminal ist nicht gerade mit einer Verarbeitung beschäftigt, die den Empfang der Meldung verhindert.

Wenn kein Benutzer am Terminal angemeldet ist (ein Log-in durchgeführt hat) oder wenn der momentane Status des Terminals den Empfang der Meldung verhindert, so reiht TAXO die Meldung in eine Warteschlange ein, bis sie übertragen werden kann. Das CM-Kommando kann auch im Batch Modus benutzt werden, um Statusmeldungen an das einleitende Terminal (von wo aus der Batch-Lauf gestartet wurde) zu senden.

10.3.1 Format des CM-Kommandos

CM
CREATE MESSAGE
RECEIVING TERMINAL NAME: [string] @
MSG: (string)

10.3.2 Benutzerantworten beim CM-Kommando

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
RECEIVING TERMINAL NAME	O	Gerätename des Terminals, an das das System die Meldung senden soll. Wird kein Terminal angegeben, so wird automatisch das benutzereigene genommen.
MSG	O	Eine einzeilige Meldung zwischen 1 und 70 Zeichen.

10.3.3 Beispiel zum CM-Kommando

Das folgende Beispiel zeigt, wie ein Benutzer das CM-Kommando eingeben muß, um an das Terminal ST02 eine Meldung abzusenden.

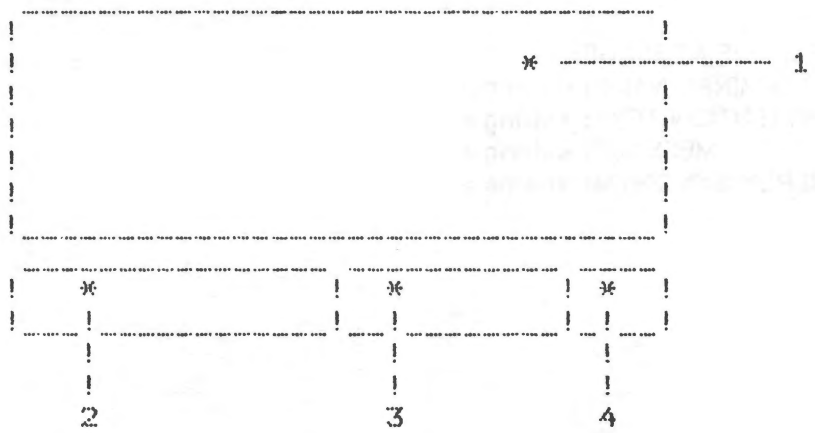
```
( ) CM
    CREATE MESSAGE
    RECEIVING TERMINAL NAME: ST02
    MSG: SYSTEM WILL BE DOWN 0900 TO 1100 6/14/78
( )
```

Eine Meldung an alle Terminals kann mit Hilfe des nachstehenden FLASH Spezialkommandos abgeschickt werden, das ein CM-Kommando an alle Terminals absetzt, die im System generiert wurden. Sorgfalt ist geboten, wenn Meldungen an alle Terminals geschickt werden sollen, da die System Table Area benötigt wird, um die Meldung festzuhalten, bis sie zum jeweiligen empfangenden Terminal gesendet werden kann.
Es folgt ein Beispiel, das die obige Meldung an die Terminals sendet. Zusätzliche Terminals können angefügt werden, indem man noch mehr CM-Kommandos an das FLASH Kommando anhängt.

```
FLASH (SYSTEM NEWS FLASH)
CM  RECEIVING = ST02
    MSG = "SYSTEM WILL BE DOWN 0900 TO 1100 4/16/78
CM  RECEIVING = ST10
    MSG = "SYSTEM WILL BE DOWN 0900 TO 1100 4/16/78
CM  RECEIVING = ST04
    MSG = "SYSTEM WILL BE DOWN 0900 TO 1100 4/16/78
```

10.3.4 SCI Anschluß (CSLM Kommando)

Der SCI-Anschluß an die Systemzeile wird durch das CSLM Kommando realisiert.
Die Systemzeile ist Bestandteil des Bildschirms am DTS. Der Bildschirm eines DTS läßt sich damit in mehrere Bereiche einteilen, die hier kurz definiert werden sollen.



		Zeile	Spalte
Bereich 1:	Benutzerbildschirmbereich	1 - 24	1 - 80
Bereich 2:	Message Bereich	25	1 - 60
Bereich 3:	Attention Meldungen	25	61 - 78
Bereich 4:	Lokale DTS Statusmeldungen	25	79 - 80

Die Bereiche 2 bis 4 heißen Systemzeile. Die Systemzeile kann nur vom System und nicht vom Benutzer unmittelbar angesprochen werden. Der Benutzer muß sich zu diesem Zweck der im folgenden beschriebenen Benutzerschnittstelle bedienen.

Anmerkung:

Die Systemzeile steht nur zur Verfügung, wenn in den DTS ein entsprechender Mikrostand eingebaut ist.

Zweck des CSLM Kommandos ist es, aus einem Batchstream Meldungen an das zugeordnete Benutzerterminal zu senden und entsprechend der Benutzerantwort den Batchstream zu beeinflussen.

Grundsätzlich gibt das CSLM Kommando zunächst den ATTENTION TEXT im Bereich 3 der Systemzeile aus. Falls der Parameter ATTENTION TEXT nicht versorgt ist, gibt das System den Standardtext MESSAGE PENDING aus. Die Meldung im Bereich 3 muß vom Bediener durch Drücken der Tastenkombination "Shift + Delete" quittiert werden. Dadurch signalisiert der Bediener dem System, daß er bereit ist, eine Mitteilung (MESSAGE) zu empfangen und zu beantworten (REPLY). Die Quittierung kann zu beliebigen Zeitpunkten während seiner Foregroundverarbeitung im Bereich 1 geschehen. Nach Quittierung der ATTENTION MESSAGE wird der Bereich 3 gelöscht und CSLM gibt den MESSAGE TEXT im Bereich 2 aus. Die Beendigung einer noch im Foreground aktiven Bildschirm EA Operation (insbes. Read, Write, Accept, Display) wird abgewartet. Nun wird eine Antwort durch den Bediener erwartet, die im REPLY SYNONYM abgelegt wird. Damit kann im Batchstream die Bedienerantwort ausgewertet werden. Während des Dialoges mit der Benutzerstation bleibt die Batchverarbeitung suspendiert.

ANMERKUNG:

Die MESSAGE und die Benutzerantwort darauf können zusammen max. 58 Zeichen lang sein.

10.3.4.1 Format des Kommandos CSLM:

```
( ) CSLM
CREATE SYSTEM LINE MESSAGE
    RECEIVING TERMINAL NAME: <acnm >
              ATTENTION TEXT: <string >
                     MESSAGE: <string >
            REPLY SYNONYM: <name >
```

10.3.4.2 Benutzerantworten zum CSLM Kommando:

Systemmeldungen	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
RECEIVING TERMINAL NAME	O	Gerätename des Terminals, an das das System die Meldung senden soll. Wird kein Gerätename angegeben, wird das benutzereigene Terminal verwendet.
ATTENTION TEXT	O	Ein Text von maximal 17 Stellen, der im Bereich 3 ausgegeben werden soll. Wird dieser Parameter nicht versorgt, so wird vom System ein Standardtext ausgegeben.
MESSAGE	O	Ein einzeiliger Text von max. 57 Stellen, der im Bereich 2 der Systemzeile ausgegeben werden soll. Wird MESSAGE nicht versorgt, muß ATTENTION TEXT angegeben werden.
REPLY SYNONYM	O	Hier ist der Name eines Synonyms anzugeben, das die Antwort aufnehmen soll. REPLY SYNONYM muß angegeben werden, wenn MESSAGE angegeben wurde.

10.3.4.3 Beispiele zum CSLM Kommando

Beispiel 1:

```
( ) CSLM
    CREATE SYSTEM LINE MESSAGE
    RECEIVING TERMINAL NAME:  ST05
              ATTENTION TEXT:  Start of Phase 2
                     MESSAGE:
               REPLY SYNONYM:
```

Der Batchstream meldet an die Station ST05, daß die Phase 2 gestartet werden soll. Der Batchstream bleibt solange suspendiert, bis die Meldung quittiert ist.

Beispiel 2:

```
( ) CSLM
    CREATE SYSTEM LINE MESSAGE
    RECEIVING TERMINAL NAME:  ST01
              ATTENTION TEXT:
                     MESSAGE:  Start of Phase 2
            REPLY SYNONYM:  A
```

Der Batchstream meldet an die Station ST01, daß eine Meldung ansteht, indem im Bereich 3 der Text "MESSAGE PENDING" ausgegeben wird. Nachdem sich der Bediener bereit erklärt hat, die Meldung zu empfangen (sh+Del), wird der Text "Start of Phase 2" im Bereich 2 der Station ST01 ausgegeben.
Der Bediener muß diese Meldung beantworten. Die Antwort wird dem Batchstream im Synonym A zur Verfügung gestellt. Anschließend wird der Batchstream fortgesetzt.

Beispiel 3:

Dieses Beispiel zeigt die Verwendung des CSLM Kommandos in einem Batch Stream, um Daten vom Bediener anzufordern, die den weiteren Ablauf des Batch Stream beeinflussen. Die einzelnen Kommandos werden anschließend erläutert.

```
-----------------------------------------------------
| * Batch Stream, um beliebige Dateien auf einem Drucker |
| * auszugeben. Der Name des Druckers sowie die Pfadna-|
| * men der Dateien werden mit Hilfe des CSLM Komman-  |
| * dos angefordert.                                   |
| *                                                    |
| * Kommando                                Erläuterung |
| *                                                    |
| BATCH                                                |
| SPDEV                                     !  ( 1 )   |
| CSLM ATTENTION = "PF Batch will was",     !  ( 2 )   |
|      MESSAGE   = "Name des Druckers",                |
|      REPLY     = DRUCKER                             |
| .LOOP                                     !  ( 3 )   |
| CSLM ATTENTION = "PF Batch will was",     !  ( 4 )   |
|      MESSAGE   = "Name der Datei",                   |
|      REPLY     = DATEI                               |
| .WHILE $DATEI   ,NE, "DATEI"              !  ( 5 )   |
|      PF FILE = $$DATEI,                   !  ( 6 )   |
|          LIST = $$DRUCKER                            |
| .REPEAT                                   !  ( 7 )   |
| EBATCH TEXT="PF Batch beendet"            !  ( 8 )   |
-----------------------------------------------------
```

Erläuterungen siehe nächste Seite!

Erläuterungen zum Beispiel 3:

(1) Das SPDEV Kommando ist notwendig, um innerhalb des Batch Streams Systemkommandos und SCI Primitivkommandos ausführen zu können.

(2) Dieses CSLM Kommando fordert vom Bediener den Namen des Druckers an, auf dem Dateien gedruckt werden sollen. Dies geschieht wie folgt:

- Im Bereich 3 des Bildschirms erscheint der Text "PF Batch will was".

- Der Bediener drückt "Shift + Delete"

- Bereich 3 wird gelöscht. Im Bereich 2 des Bildschirms erscheint nach Beendigung der aktuellen E/A Operation für den Bereich 1 der Text "Name des Druckers". Der Cursor steht hinter diesem Text. Der Bediener gibt den Namen des Druckers ein (z.B. LP01) und löst mit der Return Taste aus. Die Verarbeitung im Bereich 1 geht weiter.

- Der eingegebene Druckername wird im Synonym DRUCKER abgelegt und steht damit dem Batch Stream für die folgenden Kommandos zur Verfügung.

(3) Dieses Primitivkommando definiert den Beginn einer Schleife innerhalb des Batch Streams.

(4) Dieses CSLM Kommando fordert vom Bediener den Pfadnamen einer Datei an, die gedruckt werden soll. Dies geschieht wie folgt:

- Im Bereich 3 des Bildschirms erscheint der Text "PF Batch will was".

- Der Bediener drückt "Shift + Delete"

- Bereich 3 wird gelöscht. Im Bereich 2 des Bildschirms erscheint nach Beendigung der aktuellen E/A Operation für den Bereich 1 der Text "Name der Datei". Der Cursor steht hinter diesem Text. Der Bediener gibt den Pfadnamen einer zu druckenden Datei ein und löst mit der Return Taste aus. Die Verarbeitung im Bereich 1 geht weiter.

- Der eingegebene Dateiname wird im Synonym DATEI abgelegt und steht damit dem Batch Stream für die folgenden Kommandos zur Verfügung.

(5) Hier wird das Endekriterium für die bei (3) definierte Schleife abgefragt: Die Schleife soll beendet werden, wenn das Synonym DATEI nicht mehr definiert ist, d.h. der Synonymwert ist gleich dem Synonymnamen. Dies erreicht der Bediener, indem er keinen Dateinamen mehr bei (4) angibt, sondern nur mit der Return Taste im Bereich 2 auslöst. Ist das Endekriterium erfüllt, werden alle Kommandos bis zum Schleifenende übersprungen (siehe (7)) und der Batch wird hinter der Schleife fortgesetzt.

(6) Hier wird der Druckauftrag für den Spooler erzeugt. Die notwendigen Angaben über die auszudruckende Datei und den zu verwendenden Drucker (Pflichteingeben bei PF) werden als Synonymwerte dem Kommando PF übergeben.

(7) Dieses Primitivkommando definiert das Ende der durch (3) begonnenen Schleife.

(8) Dieses Kommando beendet die Ausführung des Batch Streams. Der Text "PF Batch beendet" wird vom Foreground SCI in der Zeile 24 des Bildschirmbereichs 1 ausgegeben, bevor das nächste SCI Kommando im Vordergrund eingegeben werden kann.

ANMERKUNG:

Da im Beispiel bei den CSLM Kommandoaufrufen kein RECEIVING TERMINAL NAME angegeben ist, werden die jeweiligen Ein/Ausgaben über die Systemzeile des Terminals abgewickelt, von dem der Batch Stream gestartet wurde.

10.4 Anzeigen von Terminalinformation (STI Kommando)

Die Benutzer können den Status ihres eigenen Terminals abfragen, indem sie das STI Kommando (Show Terminal Information) eingeben. TAXO antwortet auf die Eingabe des STI Kommandos mit der Anzeige oder Ausgabe der Meldung zur Kommandobeschreibung und der Statusinformation in folgender Form:

MOST RECENT TASK RUN ID: [Zeichenkette-1]
MOST RECENT LUNO ASSIGNMENT: [Zeichenkette-2]
TEXT EDIT IN PROGRESS: [Zeichenkette-3]
MOST RECENT EDIT FILE: [Zeichenkette-4]
TASK BEING DEBUGGED: [Zeichenkette-5]

Dabei bedeuten:

Zeichenkette-1:	Die Laufzeitidentifikation der zuletzt aktivierten Task.
Zeichenkette-2:	Die LUNO, die der Benutzer zuletzt zugewiesen hat.
Zeichenkette-3:	Y: Wenn der im SCI enthaltene Texteditor aktiv ist (Y = YES) N: Wenn der Texteditor nicht aktiv ist (N = NO)
Zeichenkette-4:	Pfadname der zuletzt editierten Datei.
Zeichenkette-5:	Task Identifikation der Task, die momentan mit dem Debugger bearbeitet wird.

10.4.1 Format des STI-Kommandos

STI
TERMINAL INFORMATION... TERMINAL:Gerätename....USER: Benutzeridentifikation

ANMERKUNG:

TAXO antwortet auf die Eingabe des STI-Kommandos mit der Anzeige einer Meldung zur Kommandobeschreibung, die den Gerätenamen des Benutzerterminals und seine Benutzeridentifikation (falls am Terminal ein Log-in verlangt wird) enthält.

10.4.2 Benutzerantworten beim STI-Kommando

Der Benutzer kann ändern:

- letzte Task-Identifikation
- letzte LUNO Zuweisung

oder

- letzter Edit File

10.4.3 Beispiel zum STI-Kommando

Das folgende Beispiel zeigt die Eingabe des STI-Kommandos durch einen Benutzer, der unter WHT016 am Terminal ST01 arbeitet.

Die angezeigte Status-Information gibt an, daß die Identifikation der Task, die zuletzt vom Terminal aus gestartet wurde, 86_{16} ist und daß eine Texteditor-Verarbeitung derzeit nicht stattfindet.

```
( ) STI
    TERMINAL INFORMATION...TERMINAL: ST01...USER: WHT06
    MOST RECENT TASK RUN ID: >86
    MOST RECENT LUNO ASSIGNMENT:
    TEXT EDIT IN PROGRESS: N
    MOST RECENT EDIT FILE:
    TASK BEING DEBUGGED:
( )
```

10.5 Anzeige des Speicherzustandes - SMS-Kommando

Benutzer können Informationen über die Nutzung des Systemspeichers und die Aktivität der SCI erhalten, indem sie das SMS-Kommando absetzen. Das Betriebssystem TAXO antwortet auf den Aufruf des SMS-Kommandos, indem es Information ausgibt über:
Speichergröße, Systemgröße, Größe der System Table Area und Nutzung des SCI.

Diese Informationen werden im folgenden Format geliefert:

```
                    SHOW MEMORY STATUS
              TOTAL MEMORY SIZE = <number> K
TAXO OS SIZE = <number> K          USER AREA = <number> K
              SYSTEM TABLE AREA = <int> WORDS
CURRENT USAGE = <int> WORDS        LARGEST AREA USED = <int> WORDS
                    SCI INFORMATION
FOREGROUND LIMIT = <int> TERMINALS   CURRENTLY ACTIVE = <int> TERMINALS
BACKGROUND LIMIT = <int> TASKS       ACTIVE/WAITING = <int>/<int> TASKS
```

10.5.1 Format des SMS-Kommandos

```
SMS
SHOW MEMORY STATUS
LISTING ACCESS NAME: [ acnm ] @
```

10.5.2 Benutzerantworten beim SMS-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
LISTING ACCESS NAME	O	Eingabe des Zugriffsnamens einer Datei oder eines Geräts, wohin die gelieferte Information ausgegeben werden soll. Standardwert ist der Terminal Local File.

10.5.3 Beispiel zum SMS-Kommando

Das folgende Beispiel zeigt die auf den Terminal Local File ausgegebenen Informationen über den Speicherstatus durch das SMS-Kommando.

```
() SMS
                    SHOW MEMORY STATUS
              TOTAL MEMORY SIZE = 128 K
TAXO OS SIZE = 0,9 K                  USER AREA = 127.1 K
              SYSTEM TABLE AREA = 9687 WORDS
CURRENT USAGE = 2385 WORDS            LARGEST AREA USED = 4533 WORDS
                    SCI INFORMATION
FOREGROUND LIMIT = 9 TERMINALS        CURRENTLY ACTIVE = 2 TERMINALS
BACKGROUND ACTIVE LIMIT = 4 TASKS     ACTIVE/WAITING = 0/0 TASKS
()
```

10.6 Ausgabe der Systemgeräte - Konfiguration (LDC-Kommando)

Um die (mit SYSGEN) in einem System generierten Geräte aufzulisten, können die Benutzer das LDC-Kommando (List Device Configuration) eingeben.

10.6.1 Format des LDC-Kommandos

```
LDC
LIST DEVICE CONFIGURATION
LISTING ACCESS NAME: [ acnm ]   @
```

10.6.2 Benutzerantworten beim LDC-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
LISTING ACCESS NAME	O	Eingabe des Zugriffnamens einer Datei oder eines Geräts, wohin die gelieferte Information ausgegeben werden soll. Standardwert ist der Terminal Local File.

10.6.3 Beispiel zum LDC-Kommando

Das folgende Beispiel zeigt die auf dem Terminal Local File ausgegebenen Informationen. Die Software Konfiguration des laufenden Systems ist die folgende:

```
() LDC
LIST DEVICE CONFIGURATION
 LISTING ACCESS NAME:    ME
   DEVICE     DEVICE TYPE     CRU/TILINE     INTERRUPT     CHASSIS     POSITION

   DS01       DISK            F800            13
   DS02       DISK            F800            13
   DS03       DISK            F820             9
   LP01       PRINTER         0060            14
   ST01       VDT             0000             6
   ST02       VDT             00C0
   ST03       VDT             0100
```

10.6.4 Anzeige des Hauptspeichers - SMM Kommando

Das SMM Kommando (Show Memory Map) wird dazu benutzt, die Belegung des Hauptspeichers anzuzeigen. Das Kommando überwacht die Verwendung des Hauptspeichers durch das System und die Benutzer-Tasks.

10.6.4.1 Format des SMM Kommandos

SMM
SHOW MEMORY MAP

10.6.4.2 Benutzerantworten zum SMM Kommando

Keine erforderlich

10.6.4.3 Beispiel zum SMM Kommando

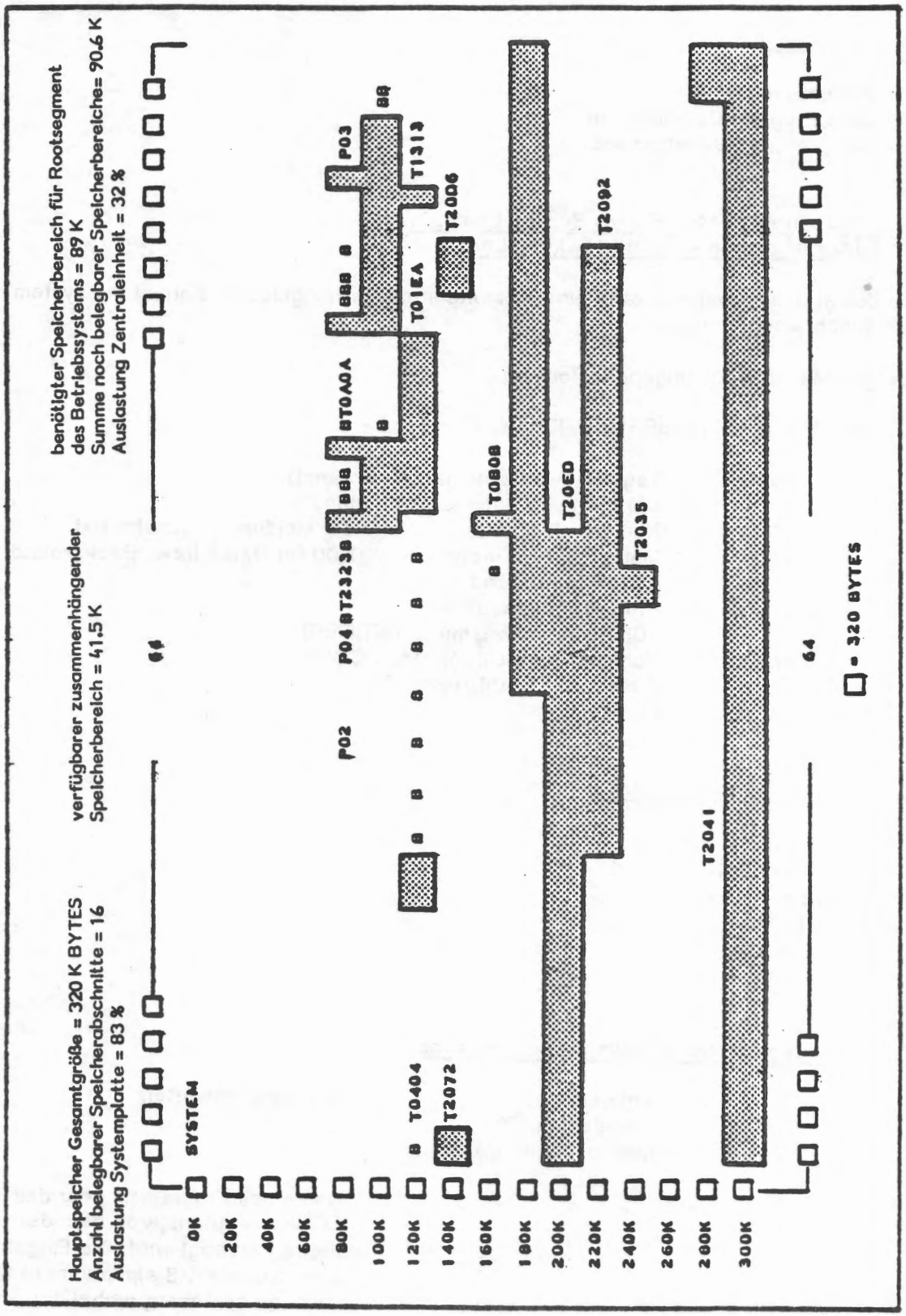
Das folgende Beispiel zeigt den Gebrauch des SSM Kommandos.

Der im linken oberen Teil des Bildschirms gezeigte Wert (Memory Size) gibt die Gesamtgröße des für das System installierten Hauptspeichers in K-Byte an (1 K-Byte = 1024 Byte). Die Angabe "Static Memory" zeigt die Größe des für das Rootsegment des Betriebssystems benötigten Speicherbereichs, sowie die für die speicherresidenten Prozeduren und speicherresidenten Tasks benötigten Speicherbereiche an. Dieser Wert wird ebenfalls in K-Byte angezeigt.

Die zweite Bildschirmzeile gibt Informationen über freie Speicherbereiche. Die Zahl, die neben dem Text "# OF HOLES" steht, gibt die Anzahl noch belegbarer Speicherabschnitte an. Die Angabe "LARGEST BLOCK" gibt den größten verfügbaren, zusammenhängenden Speicherbereich an. Der rechte Teil der zweiten Zeile gibt die Summe aller noch belegbaren Speicherbereiche an.

Die dritte Bildschirmzeile zeigt eine Benutzungsstatistik für die Systemplatte (links) und die Zentraleinheit (rechts). Diese Information wird alle 4 Sekunden auf den neuesten Stand gebracht und gibt die Zeit in Prozent an, während der in den vorangegangenen 4 Sekunden der Hauptspeicher beansprucht wurde. Die Anzeigen des SMM Kommandos selbst benötigen einen kleinen Teil der angegebenen CPU-Zeit. Die restlichen Zeilen der Anzeige geben eine schematische Darstellung des Hauptspeichers wieder. Die niedrigste Speicheradresse erscheint oben, die höchste unten am Bildschirm. Dieser Teil der Anzeige besteht aus 16 Zeilen zu je 64 grafischen Zeichenpositionen. Die durch jede Position dargestellte Speichergröße wird ganz unten am Bildschirm angezeigt. Als Teil der Anzeige zeigt das Betriebssystem auch die Puffer und die Tasksegmente an. Jedes mit einem hervorgehobenen P beginnende Segment ist ein Prozedursegment. Die zweistellige Zahl, die hinter dem P erscheint, ist das installierte Identifikationszeichen (ID) der Prozedur. Jedes mit einem hervorgehobenen T beginnende Segment ist ein Tasksegment. Die ersten zwei der hinter dem T angezeigten Ziffern ist das installierte Identifikationszeichen (ID) der Task. Die letzten zwei Ziffern des Tasksegments geben die Run-time ID der Task an. Jedes mit einem hervorgehobenen B beginnende Segment ist ein Blockpuffer der Dateiverwaltungsroutine. Leere Bildschirmbereiche gehören zur Überschrift links der ersten Leerstelle. Das dunkle grafische Symbol zeigt die freien Speicherbereiche an.

Beispiel zum SMM Kommando



10.7 Allgemeine Kommandos

Das folgende Kapitel enthält Kommandos von allgemeiner Bedeutung. Hierunter fallen

10.7.1 Die Kommandos

SLG - System Log
SPDEV - Start Program Development
QPDEV - Quit Program Development

10.7.2 Eintragen einer Meldung ins System Logbuch (SLG Kommando - Write to System Log)

Mit Hilfe des SLG-Kommandos kann eine Meldung in einem festgelegten Format ins System Logbuch geschrieben werden.

Die erzeugte Meldung hat folgendes Format:

DDD:ZZZZbbbbbbSTNObMObPROGR-IDbXXXXb <TEXT>

wobei	DDD	Tag-Nr. (wird automatisch ergänzt)
	ZZZZ	Uhrzeit (wird automatisch ergänzt)
	STNO	Nummer der Station, welche die Meldung abgesetzt hat
	MO	Mode IOF für Foreground VDT,00 für Batch bzw. Background, 01 für Foreground TTY
	PROGR-ID	Programmname oder SCI-Kommandoname (IDENTIFIER)
	XXXX	Nummer der Meldung (MSG CODE)
	(Text)	Zusatztext (wahlweise) (TEXT)
	b	Leerzeichen

10.7.2.1 SLG-Kommando-Format

SLG
WRITE TO SYSTEM LOG
    IDENTIFIER: <name>
    MSG CODE: <int>
        TEXT: <string>

10.7.2.2 Benutzerantworten beim SLG-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
IDENTIFIER	R	Name des Programms oder des SCI-Kommandos, von dem der Eintrag erzeugt wird. Die Eingabe kann aus max. 8 alphanumerischen Zeichen bestehen, wobei das erste Zeichen ein Buchstabe sein muß.

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantworten
MSG CODE	R	eine 4stellige Ganzzahl, die die Meldung spezifiziert. Der Code kann vom Anwendungspaket festgelegt werden. Zur Zeit werden vom System nur folgende Codes verwendet: 0100 KZ f. Login 0199 KZ f. Logoff
TEXT	O	Zusatztext max. 45 Zeichen; hier kann eine beliebige Zeichenkette angegeben werden, die die Meldung genauer beschreibt.

10.7.2.3 SLG-Kommando Beispiel

Das folgende Beispiel zeigt, wie eine Meldung mit SLG ins Logbuch geschrieben werden kann:

```
( ) SLG
WRITE TO SYSTEM LOG
    IDENTIFIER: SCI
    MSG CODE: 0100
    TEXT: ST02 M = 0F LOG IN

( ) SF
SHOW FILE
    FILE PATHNAME: .S$SLG1
                    :
                    :
                    :
189: 1507 MSG GGE022 SCI  0100 ST02 M=0F LOG IN
```

10.7.3 Zugriff auf die allgemeinen Systemkommandos SPDEV - Start Program Development Kommando

Normalerweise können die für einen Anwender verfügbaren SCI-Kommandos in 2 Gruppen eingeteilt werden:

1. Kommandos für benutzerspezifische Anwendungsprogrammsysteme.

2. Kommandos allgemeiner Art zur Systempflege und Programmentwicklung.

Die in diesem Bedienerhandbuch beschriebenen Kommandos sind größtenteils der zweiten Gruppe zuzuordnen.

10.7.3.1 SPDEV-Kommando (Start Program Development)

Da ab TAXO V 2.0.0 die Kommandobibliotheken für Anwendung und Programmentwicklung nicht mehr getrennt sind. (.TA$PROL ist ALIAS Name für .S$PROC), erfolgt keine Umschaltung mehr. Im Foreground ist jedoch nach SPDEV die Eingabe von Primitivkommandos durch den Benutzer möglich. Zusätzlich wird das SCI Kommando Prompting auf

PDEV V 201 ( )

gesetzt, um dem Benutzer anzuzeigen, daß nun alle Möglichkeiten der Programmentwicklung zur Verfügung stehen. Im Background/Batch wird SPDEV ignoriert. Das Kommando SPDEV benötigt keine Parameter.

10.7.3.2 QPDEV-Kommando (Quit Program Development)

Da ab TAXO V 2.00 die Kommandobibliotheken für Anwendung und Programmentwicklung nicht mehr getrennt sind. (.TA$PROC ist ALIAS Name für .S$PROC), erfolgt keine Umschaltung mehr. Im Foreground ist jedoch nach QPDEV die Eingabe von Primitivkommandos durch den Benutzer gesperrt. Zusätzlich wird das SCI Kommando Prompting auf

SCI V 200 ( )

gesetzt, um dem Benutzer anzuzeigen, daß er sich in der Standard Kommandobibliothek befindet. Im Background/Batch wird QPDEV ignoriert. Das Kommando QPDEV benötigt keine Parameter.

10.7.4 SHOW DTS LOG (SDTSL-Kommando)

Bei der Systemgenerierung wird pro DTS ein PDT (Physical Device Table) angelegt. Pro DTS spezifischen PDT werden 12 Bytes für unterschiedliche Fehlermeldungen angelegt. Diese Fehlerzellen werden immer dann aufgezählt, wenn ein Fehler während der Datenübertragung zwischen DTS und Zentraleinheit auftritt. Alle DTS spezifischen Fehlerzellen werden immer nach dem Ladevorgang des TAXO Betriebssystems gelöscht. Die eingetragenen Werte bleiben solange erhalten, bis das Betriebssystem erneut geladen wird. Der Aufzählvorgang erfolgt bis max. 255 und wird anschließend eingestellt, es erfolgt keine Moduloverrechnung.

Mit dem SCI Kommando SDTSL können alle Fehlerzellen für alle DTS abgefragt werden. Die Anzeige oder Ausgabe erfolgt entsprechend des eingegebenen OUTPUT ACCESS NAME. Die Auswertung der einzelnen Fehlerbytes erfolgt in dezimaler Darstellung.

10.7.4.1 Format des SDTSL Kommandos

SDTSL
SHOW DTS LOG
OUTPUT ACCESS NAME: <acnm> @

10.7.4.2 Benutzerantworten beim SDTSL-Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
OUTPUT ACCESS NAME:	O	Ein Gerätename oder Pathname einer File. Standardwert ist die Terminal Local File des Anwender Terminals.

10.7.4.3 SDTSL Kommando Beispiel

```
( ) SDTSL
    SHOW DTS LOG
    OUTPUT ACCESS NAME: .SEG003.TEST
( )
```

10.7.5 Clear Secret Synonyms (Q$SYN-Kommando)

Wenn bei der Synonymtabelle ein Overflow-Fehler auftritt, sind zu viele Synonyme definiert. Der Tabellenbereich für die vom System definierten verborgenen Synonyme und die Benutzersynonyme ist identisch. Die vom System definierten verborgenen Synonyme können durch Aufruf von Q$SYN gelöscht werden. Das Kommando SF (Show File) der Datei .S$PROC.Q$SYN zeigt die mit Q$SYN gelöschten Synonyme.

10.7.6 COBOL Anschluß

Der COBOL Anschluß ist im COBOL Runtime realisiert. Das COBOL Sprachelement ist "STOP <Literal>". Das Literal wird auf der Systemzeile (Bereich 3) ausgegeben. Die betreffende COBOL Task wird suspendiert. Der Bediener muß die Meldung durch die Tastenfolge "Shift + Delete" quittieren. Die Quittierung kann zu beliebigen Zeitpunkten während einer Foreground-verarbeitung im Bereich 1 geschehen. Nach der Quittierung der Meldung wird die COBOL Task wieder fortgesetzt.

10.7.7 Unterstützung länderspezifischer Zeichenvorräte (Country Code) für Key Indexed Dateien.

10.7.7.1 Allgemeines

Das TAXO Betriebssystem verwendet beim Einfügen von Datensätzen in Key Indexed Dateien normalerweise die Sortierreihenfolge nach dem Standard ASCII Code. Dies kann bei Textdaten, die landesspezifische Zeichen enthalten, zu einer für das Land falschen alphabetischen Sortierung führen.

Beispiel:

Sortierreihenfolge der Begriffe ANNA, ÄNNCHEN, BERTA

bei deutscher Sortierung:	ANNA ÄNNCHEN BERTA	richtig
internationaler (US) Sortierung:	ANNA BERTA ÄNNCHEN	falsch

Um dieses Problem bei Key Indexed Dateien zu beseitigen, erlaubt TAXO dem Benutzer, eine vom Standard ASCII Code abweichende Sortierreihenfolge zu spezifizieren. Die eingestellte Sortierreihenfolge wird als Country Code im System abgelegt und gilt global für alle Key Indexed Dateien.

Der für ein Betriebssystem gültige Country Code wird normalerweise bei der Systemgenerierung festgelegt. Er kann dann dynamisch zur Laufzeit durch das SCI Kommando MCC (Modify Country Codes) geändert werden. Außerdem besteht durch das SCI Kommando SCC (Show Country Code) die Möglichkeit, den aktuell im System gültigen Country Code abzufragen. Die Tabelle a zeigt die von TAXO unterstützten Country Codes.

Anmerkung:
Der für ein System eingestellte Country Code hat nur Einfluß auf die Sortierreihenfolge bei Key Indexed Dateien. Er hat keine Wirkung auf Sortiervorgänge, die mit dem Sort/Merge Dienstprogramm durchgeführt werden. Soll hierbei eine vom Standard ASCII Code abweichende Sortierreihenfolge verwendet werden, so muß diese durch spezielle Sort/Merge Steueranweisungen definiert werden.

10.7.7.2 Anzeigen des im System eingestellten Country Codes - Kommando SCC (Show Country Code).

Das Kommando SCC (Show Country Code) ermöglicht es einem Benutzer, den im System, aktuell gültigen Country Code (landesspezifische Sortierreihenfolge) in eine Datei oder auf ein Gerät ausgeben zu lassen. Die Ausgabe enthält die (englischen) Ländernamen, in denen der eingestellte Code gilt.

10.7.7.2.1 Format des SCC Kommandos

SCC (SHOW COUNTRY CODE)
OUTPUT ACCESS NAME: [acnm] @

10.7.7.2.2 Benutzerantworten zum SCC Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
OUTPUT ACCESS NAME	O	Der Zugriffsname eines Gerätes oder einer Datei, wo die Ausgabe des SCC Kommandos erfolgen soll. Wird keine Antwort eingegeben, wird der Terminal Local File des Benutzerterminals verwendet.

10.7.7.2.3 Beispiel zum SCC Kommando

Beispiel 1: Im System ist der Standard ASCII Code eingestellt (Voreinstellung in allen TAXO Standard Betriebssystemen)

```
( ) SCC
    SHOW COUNTRY CODE
    OUTPUT ACCESS NAME:

        UNITED STATES
```

Beispiel 2: Im System ist der deutsche Zeichensatz eingestellt.

```
( ) SCC
    SHOW COUNTRY CODE
    OUTPUT ACCESS NAME:

        GERMANY/AUSTRIA
```

10.7.7.3 Ändern des im System eingestellten Country Codes - das Kommando MCC (Modify Country Code)

Das Kommando MCC (Modify Country Code) ermöglicht es einem Benutzer, den im System eingestellten Country Code (landesspezifische Sortierreihenfolge) zu ändern. Der nach der Kommandoausführung gültige Country Code wird im Protokoll ausgegeben.

Warnungen:

1. Bei Key Indexed Dateien können durch das Arbeiten mit verschiedenen Country Codes Inkonsistenzen bis hin zur völligen Zerstörung der Datei auftreten.

   Deshalb:

   Key Indexed Dateien immer mit dem gleichen Country Code verarbeiten.

2. Binärdaten werden - je nach Country Code - in anderer Reihenfolge sortiert.

Zur Zeit stehen folgende landesspezifische Country Codes zur Verfügung, die beim Kommandoaufruf angegeben werden können:

Tabelle a : Verfügbare Country Codes

Country Code	Land
USA	USA - United States of America
UK ENGLAND	England- United Kingdom
JAPAN	Japan
FR BE	Frankreich - Belgien
DEN NOR	Dänemark - Norwegen
AR	Arabische Länder
GER	Deutschland (Germany) -
AU	Österreich (Austria)
SW FI	Schweden - Finnland

10.7.7.3.1 Format des MCC Kommandos

MCC MODIFY COUNTRY CODE)
COUNTRY NAME: <string> @
OUTPUT ACCESS NAME: [acnm] @

10.7.7.3.2 Benutzerantworten zum MCC Kommando

Systemmeldung	Antwort erforderlich (R) oder optional (O)	Benutzerantwort
COUNTRY NAME	R	Einer der in Tabelle a angegebenen Country Codes.
OUTPUT ACCESS NAME	O	Der Name eines Geräts oder einer Datei, wo der nach der Kommandoausführung gültige Country Code ausgegeben wird. Wird keine Eingabe gemacht, wird der Terminal Local File des Benutzerterminals verwendet.

10.7.7.3.3 Beispiel zum MCC Kommando

Es soll die deutsche Sortierreihenfolge eingestellt werden.

```
( ) MCC
    MODIFY COUNTRY CODE
    COUNTRY NAME: GER
    OUTPUT ACCESS NAME:
    GERMANY/AUSTRIA
( )
```

10.7.8 Anzeigen von Fehlermeldungen am Benutzerterminal

Durch ein neues Hilfsprogramm und einige neue SCI-Kommandos ist es möglich, ausführliche Fehlertexte auf einer TAXO Systemplatte abzulegen und sie bei Bedarf am Benutzerterminal anzuzeigen. Da die Fehlertexte in Key Indexed Dateien abgelegt werden, dafür aber ein großer Platzbedarf notwendig ist, sind die zur Zeit vorhandenen Fehlertexte nur in sequentiellen Dateien auf der Standard TAXO Systemplatte vorhanden. Im Directory .S$ERROR der Systemplatte befinden sich zwei Unterdirectories DOCU und SEQ. Im Unterdirectory DOCU befindet sich in der Datei DEUTSCH eine deutsche, in der Datei ENGLISH eine englische Beschreibung über das Fehleranzeigesystem. Diese Beschreibungen (Pfadname also .S$ERROR.DOCU. DEUTSCH bzw. .S$ERROR.DOCU.ENGLISH) können mit PF (Print File) unter Angabe von ANSI FORMAT = YES ausgedruckt werden.

Das Unterdirectory SEQ enthält eine Reihe von sequentiellen Dateien, in denen die zur Zeit vorhandenen Fehlertexte abgelegt sind. Die erwähnten Beschreibungen enthalten Hinweise, wie von sequentiellen Dateien in die entsprechenden Key Indexed Dateien kopiert werden können. Die vorhandenen Fehlertexte sind in Englisch abgefaßt und erheben keinen Anspruch auf Vollständigkeit. Wir möchte vielmehr alle TAXO Anwender einladen, die Fehlertexte entsprechend ihren Anforderungen zu ändern bzw. zu erweitern.

11 GEBRAUCH DES LINK EDITORS

11.1 Merkmale

Der Link Editor wird dazu verwendet, um einzelne Objektmodule zu einem unter TAXO laufenden Programm zusammenzubinden.

Das plattenorientierte Betriebssystem TAXO ist ein Multitasking Betriebssystem und unterstützt alle Eigenschaften des Link Editors. Da es ein plattenresidentes System ist, ist es gut geeignet für Programme, die in Overlays eingeteilt sind; es unterstützt außerdem automatisches Laden von Overlays - siehe Beschreibung im Link Editor Handbuch.

Die folgenden Eigenschaften des Link Editors werden durch TAXO unterstützt:

- Automatisches Laden von Overlays
- Direktzugriffs-Bibliotheken
- Sequentielle Bibliotheken
- Linken von COBOL Programmen
- Formate: ASCII, Compressed und Image
- Absolute Speicherpartionierung

Mehr Information zu diesen Eigenschaften kann aus dem Link Editor Handbuch entnommen werden.

11.2 Arbeitsweise des Link Editors

Der erste Schritt zur Durchführung eines Linkeditorlaufs ist, eine Steuerdatei zu entwickeln, die die Linkeditorfunktionen definiert. Die Steuerdatei kann mit Hilfe des TAXO Texteditors angelegt werden.
Die Steuerdatei enthält sowohl die Kommandos des Linkeditors als auch die Namen der Objektmodule.

Der Link Editor wird ausgeführt, indem man an einer Station das Kommando XLE eingibt. Man beachte, daß die Station im Kommandomodus ist, bevor das Kommando eingegeben wird. Wurde XLE eingetastet, so wird folgende Bildschirmanzeige am VDT ausgegeben:

```
EXECUTE LINKAGE EDITOR
  CONTROL ACCESS NAME:
  LINKED OUTPUT ACCESS:
    LISTING ACCESS NAME:
            PRINT WIDTH:   80
```

Als Antwort auf die Ausgabe CONTROL ACCESS NAME: muß der Benutzer den Pfadnamen des Gerätes oder der Datei angeben, von der die Steueranweisungen (control stream) gelesen werden sollen. Die Steuerdatei kann eine sequentielle Plattendatei sein oder jedes andere sequentielle Gerät wie Bandeinheit. Die folgende Zeile zeigt als Beispiel den Eintrag des Pfadnamens einer sequentiellen Plattendatei:

```
CONTROL ACCESS NAME:    VOL2.EDITOR.CONFILE
```

Man beachte, daß es keinen Standardwert für CONTROL ACCESS NAME: gibt. Ein Übergehen dieser Eingabe ist nicht möglich.

Als Antwort auf die LINKED OUTPUT ACCESS NAME Anzeige gibt der Benutzer den Pfadnamen des/der sequentiellen Gerätes/Datei ein, auf das/die die Ausgabe des Linkeditors geschrieben werden soll. Wird keine Objektausgabe gewünscht, so kann der Bediener "DUMY" eingeben, wodurch die Generierung der Ausgabe unterdrückt wird. Der Gebrauch von DUMY erlaubt dem Benutzer einen Probelauf, um sicher zu gehen, daß keine Fehler auftreten. Im Folgenden sieht man als Beispiel die Eingabe eines Pfadnamens einer sequentiellen Plattendatei:

```
LINKED OUTPUT ACCESS NAME:    VOL2.LINK.OUT1
```

Wird im FORMAT Kommando die IMAGE Option gewählt, so muß der Eintrag, der auf LINKED OUTPUT ACCESS NAME eingegeben wurde, eine TAXO Programmdatei oder eine TAXO System Image Datei sein.

Als Antwort auf die Ausgabe LISTING ACCESS NAME gibt der Bediener das Gerät bzw. die Datei an, wo das Protokoll des Link Editor Laufes erstellt werden soll. Wird diese Liste nicht gewünscht, so kann der Benutzer "DUMY" ausgeben, wodurch die Generierung dieser Ausgabe unterdrückt wird. Der als Antwort auf diese Anzeige eingegebene Wert kann jeder gültige TAXO Zugriffsname, Synonym oder Gerätename sein. Das folgende Beispiel bewirkt die Ausgabe der Liste am Zeilendrucker.

```
LISTING ACCESS NAME:    LP01
```

Zur Beschreibung des Link Editor Listings siehe das Link Editor Handbuch.

Die letzte Anzeige PRINT WIDTH: erlaubt dem Benutzer die Länge der zu druckenden Zeile anzugeben, oder als Standardwert, den Wert 80 Zeichen zu akzeptieren.

Das folgende Beispiel zeigt die Antworten auf die Anforderungen, wobei die Steuerdatei auf VOL1.EDITOR.CONFILE ist, das Gerät für die Liste der Zeilendrucker eins (LP01) ist, und der Standardwert für PRINT WIDTH angenommen wurde:

```
XLE
  EXECUTE LINKAGE EDITOR
   CONTROL ACCESS NAME:    VOL1.EDITOR.CONFILE
LINKED OUTPUT ACCESS NAME: VOL1.LINK.OUT1
      LISTING ACCESS NAME: LP01
               PRINT WIDTH: 80
```

12 INSTALLIEREN, ANZEIGEN, VERÄNDERN UND LÖSCHEN VON PROGRAMMEN IN PROGRAMMDATEIEN

12.1 Einführung

Unter dem TAXO Betriebssystem lauffähige Programme können aus mehreren Teilen bestehen:

- eine oder zwei mehrfach verwendbare (reentrante) Prozeduren. Prozeduren können Bestandteil von mehreren verschiedenen Programmen sein. Ein Programm muß nicht unbedingt Prozeduren enthalten.
- einer Task. Ein Programm muß mindestens eine Task enthalten.
- ein oder mehrere Overlays. Dies sind Programmteile, die zur Laufzeit eines Programms bei Bedarf in einen freigehaltenen Speicherbereich der Task von der Platte in den Hauptspeicher nachgeladen werden können. Ein Programm muß nicht unbedingt Overlays enthalten.

Weitere Informationen über den Aufbau von Programmen können dem Handbuch des TAXO Link Editors entnommen werden.

Die verschiedenen Teile eines Programms (Prozeduren, Task, Overlays) werden als Module oder Segmente des Programms bezeichnet. Da ein unter TAXO lauffähiges Programm mindestens aus dem Taskmodul bestehen muß, wird ein Programm auch als Task bezeichnet. Beim Starten eines Programms spricht man deshalb auch vom Starten einer Task (siehe z.B. das Kommando XT - Execute Task u.ä. im Kapitel 4.2).

Vor der Ausführung eines Programms müssen alle seine Module in einer Programmdatei installiert werden. Dies ist auf zwei Arten möglich:

- Durch den TAXO Link Editor unter Verwendung des IMAGE Formats für die Ausgabe des Objektcodes des Programms (siehe Handbuch des TAXO Link Editors).
- Durch die in diesem Kapitel beschriebenen Installationskommandos.

Nach der Installation der Module eines Programms können diese angezeigt, verändert und wieder gelöscht werden. Die hierfür zur Verfügung stehenden Kommandos werden ebenfalls in diesem Kapitel beschrieben.

Für das Verständnis der in diesem Kapitel beschriebenen Kommandos ist es notwendig, einige Kenntnisse über den Aufbau von Programmdateien unter dem TAXO Betriebssystem zu besitzen. Diese sollen im nächsten Abschnitt vermittelt werden. Weitere detaillierte Informationen können dem Handbuch "TAXO Dateistrukturen" entnommen werden.